实时应用中的鲁棒车道检测与跟踪

https://github.com/BlueBubbleWei/Lane-Detection

**摘要:**有效的车道检测算法是先进驾驶员辅助系统的重要组成部分，它为行车安全提供了重要信息。车道检测和跟踪算法所面临的挑战包括：车道标线缺乏清晰度、恶劣天气能见度低、光线和光线反射、阴影以及密集的道路指示。本文提出了一种鲁棒的、实时的基于感兴趣区域的车道检测算法，以降低高噪声水平和计算时间。该算法同时处理梯度线索和颜色线索，并用扫描线进行线聚类，验证车道标志的特征。它使用累积的统计数据去除任何假车道标记并跟踪真实的车道标志。实验结果表明，该算法计算结果准确，满足了低功耗嵌入式系统的实时操作要求。

指数条款-车道车道线检测，跟踪，实时聚类，卡尔曼滤波器，感兴趣区域

1. **引言：**近年来，对智能车辆技术的研究日益深入。特别是，对于高级驾驶员辅助系统（ADAS）技术已被广泛应用于商用车在安全自主驾驶的兴趣。车道线检测功能的司机如车道结构和车道其他车辆的位置，提供基本的信息在系统中起着重要的作用。检测结果应准确及时提供。

摄像机的视觉数据由于其信息含量高、成本低和操作能力强，以及在必要时使用超声波传感器或雷达作为辅助传感器而成为最常见的方式[ 1 ]。在基于视觉的车道检测方法中，通过图像处理分析从摄像机（通常安装在前挡风玻璃后面以获取车道信息）捕获的二维图像。基于视觉的方法可以分为两类：基于特征的模型和基于模型的方法。

基于模型的方法通常使用带有参数的数学模型来描述车道结构[ 2 ] [ 4 ]。虽然已经证明基于模型的方法比基于特征的方法对噪声更具鲁棒性，但由于它们需要一些已知的几何参数和繁重的计算，所以它们很难实现。

基于特征的方法分析图像和检测像素信息的梯度或图案的颜色来识别车道标记[ 5 ] [ 9 ]。大冢[ 5 ]利用边缘图像，将噪声通过直方图分析和检测边缘点的方向找到车道边界。daigavane [ 6 ]开发了一个蚁群优化（ACO）环节脱节的边缘应加入；他们用Hough变换提取车道标线。基于特征的方法的另一种方法利用图像中的颜色信息[ 7 ] [ 9 ]。颜色分割的彩色模型如HSV和YUV应用后，取而代之的是RGB模式，像素分为车道标线类或道路背景类。使用梯度信息的方法要求清晰的车道标志，很容易受到阴影和闭塞，而使用颜色信息的方法受影响较小。然而，颜色的一致性取决于光照、光反射和传感器类型。

算法的低复杂度也需要在车辆嵌入式系统上实现实时操作。大多数以前的工作不符合要求，不能申请现实世界的应用程序，除非车辆携带一台PC机。

本文提出了一种鲁棒快速的车道检测与跟踪算法，该算法由初始化、车道检测和车道跟踪三个步骤组成。车道检测和车道跟踪计划是复杂的，包括许多新的功能，如彩色车道检测，线聚类，扫描线测试，和统计信息的测试与作者以前的工作相比。该算法利用扫描线法对车道标线进行验证，以确定检测车道标志的有效性。它使用累积的统计数据去除假车道标记并跟踪真实的车道标志。改进的车道跟踪方法减少了噪声线段，提高了检测率。实验结果表明，该算法在各种不同类型的车道标识，以及在有问题的天气条件下，如雨雪环境下，是非常健壮的。实现结果表明，该算法在低计算能力的嵌入式系统上能实时工作。

**II提出的算法：**所提出的算法包括初始化、车道标记检测和车道标记跟踪，如图1（a）所示。车道标志跟踪是主要的操作过程，而初始化和车道标识检测则建立了矩形ROI和ROI。

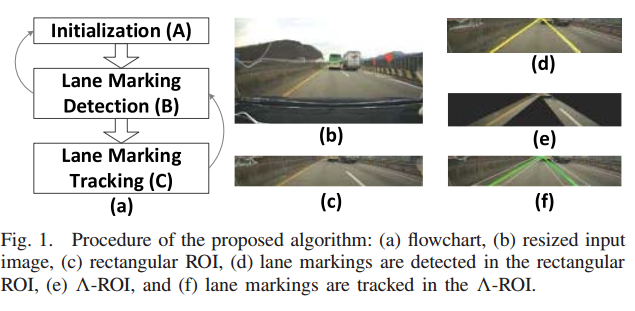


图1。提出的算法程序的flowchart：（a），（b）resized输入图像的矩形ROI（C），（D）是detected markings车道的矩形ROI，ROI（e）和（f）是tracked markings车道的ROI。

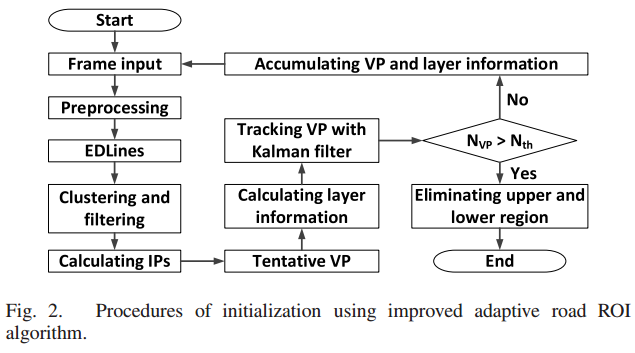


图2。基于改进自适应道路ROI算法的初始化过程

1. **初始化：**在初始阶段配置矩形ROI，并且仅在移除消失点上方的区域和与摄像机相连的车辆的较低区域时包括道路区域。基于自适应道路ROI确定算法的改进算法确定矩形ROI（10）。首先对输入图像进行预处理，如图2所示。由于图像的分辨率大于某个值，则车道检测结果几乎相同。在缩小过程中，彩色图像被转换为灰度图像。

边缘使用edlines方法[ 11 ]原始灰度图像中提取。去掉比阈值短的边，并将幸存的边聚集成线段。为了保持与车道标记平行的线段，采用斜率滤波器。图像分为右半区（RHR）和左半区（LHR）使用垂直线。在RHR线段和线段的LHR选择得到一个交叉点（IP）通过延长线段。交叉点是从所有的线段对中计算出来的，大部分是在实际的VP附近找到的，因为大多数幸存线段几乎是平行的。暂定的VPS和层信息的积累和利用Kalman滤波器进行一定数量的帧，直到有效的VPS数跟踪（NVP）变得大于阈值数（n）。

改进的自适应道路ROI算法比以前的算法（10）更好地确定了矩形ROI。以前的算法偶尔会发现一个假VP时，从噪声的线段数与实际车道标记相媲美。此外，当照明罩上的线段与照明时的车道标志平行时，也不会移除车辆的引擎罩。改进的自适应道路ROI算法对所有测试视频都正确地检测出矩形ROI，而上一算法的平均检测率为94%。

**B标线检测:**图3显示了矩形ROI中检测到的车道标记的流程图。确定了矩形ROI后，对该区域的车道标志进行了搜索。将矩形ROI的彩色图像转化为灰度图像。一种由HSV颜色模型中的值（v）组成的像素组成的像素。

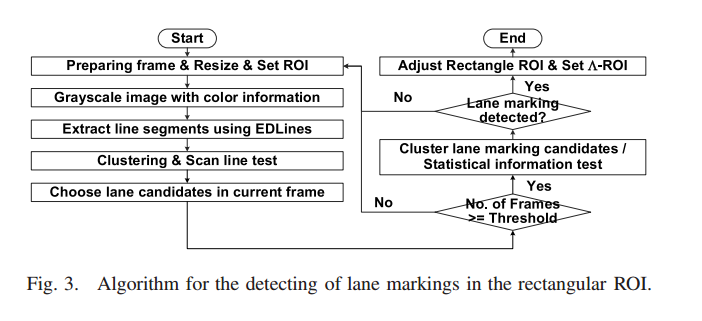


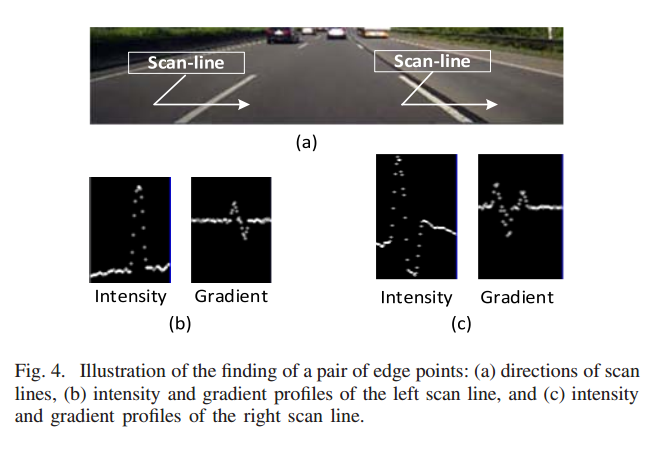
图3。矩形ROI的车道标识检测算法。

edlines方案提取线段从灰度图像[ 11 ]。的edlines优于Canny边缘检测和PPHT，在精度和性能方面，我们在以前的算法[ 12 ]使用；此外，它增加了检测率和减少计算时间。线段可以通过斜率滤波器进行滤波，因为车道标记的线段指向VP并具有有限的斜率值。通过斜坡滤波器的线段与车道标志平行。矩形ROI分为LHR的垂直线，通过形象的VP RHR，和过滤的斜坡在LHR和RHR不同。斜率滤波器有效地去除许多噪声线段。[ 12 ]。

对于道路背景使用白色车道标志的车道标线检测，灰度信息很有用；然而，由于低对比度，使用灰度信息很难检测到彩色车道标志（例如黄色或蓝色），特别是当彩色车道标记不清晰时。该算法主要包括基于YUV颜色模型的彩色车道标线检测的自适应方法，彩色车道标线是在YUV颜色空间[ 13 ]检测更加容易。黄色和蓝色的车道标线的颜色差异的图像具有较高的对比和色差信息添加到灰度图像，车道标线图案放大；这增加的车道标线的检测率。然而，如果在夜间或在黄色灯光的隧道中，背景颜色出现黄色，彩色车道标线图案在彩色图像中不清晰。在这种情况下，颜色信息不添加到灰度图像中，因为它增加了灰度图像中的噪声分量。

车道标志是一条漆得很亮很粗的线，它可以根据车道标志的厚度、弯曲的车道标线和模糊不清的图像产生两条以上的边。显然，从车道标志的边界中提取的线段与斜率和位置之间具有相似性，这些端点是通过端点位置获得的。此功能可用于区分来自车道标线和来自噪声的线段。分层聚类（HAC）方法应用到集群的线段，线段从单车道合并成一个单一的线段。合并过程由垂直聚类和水平聚类组成。垂直聚类合并从车道标志的同一侧提取的线段，水平聚类将从车道标志的相对侧合并线段。合并的线段有一个大于1的权重因子，这取决于水平合并的数量。

由于受到诸如裂缝、新路面、阴影和其他车辆等噪音所引起的线段的影响，在聚类之后仍存在若干线段。车道标志的独特之处是它们比背景（或道路）亮度更大，而且它们在相似的位置上连续出现。如果通过合并的簇绘制水平线并沿着线扫描强度，则可以获得低、高、低强度图案，如图4所示。强度的导数在车道标志的边缘点上表现为正峰和负峰。通过比较两个峰值内的像素的平均强度和两个峰外的像素的平均强度，可以利用强度模式及其衍生物来验证线段是否从车道标记中提取出来。如果两个强度之间的差异高于某一特定阈值，则线段满足强度条件。上述车道标志的验证过程，称为“扫描线测试”，用于探测车道标志。该测试扫描越过线段的虚拟水平线，以在真实的车道标记上找到边缘点对，如图4（a）所示。扫描线测试开始于穿过低端点的虚拟水平线，并且沿着向上端点的等距水平线重复。如果幸存线段通过指定数量的扫描线测试，则它被认为是当前帧中真正的标记候选线。收集线段以进行进一步的处理步骤。在道路状况很差的情况下，可能没有一条线段满足扫描线测试；例如，车道标记被侵蚀，它们会受到夜间道路上光线反射的影响，或者它们不在道路边界上绘制。

图4。关于一对边缘点的发现：（a）扫描线的方向，（b）左扫描线的强度和梯度分布图，（c）右扫描线的强度和梯度分布图。

从其他车辆，阴影或建筑物的线段可以生存的斜率滤波和聚类，如果它们是平行的车道标志。在正常行驶条件下，连续帧中车道标志的位置变化不大。真正的车道标记的线段将连续出现在有限的范围内，而那些来自噪声的线段只会出现在几个连续的帧中，或者位置不同。该算法的车道标线积累一定数量的帧的候选人（例如，30），这群考生在相似的位置，这是所谓的“帧间聚类。”为帧间聚类算法的边坡，积累统计数据，较低的IPS和VPS的位置，和车道宽度。该算法检查车道宽度（对称性）和较低的IP位置（相似条件）松为车道线的考生，选择从LHR和RHR的矩形ROI，分别对对，对车道线考生应在一个类似的副总裁的位置对上层IP（VP条件）。对所有的对重复测试。

V-ROI 的回报率是建立在每个线段在RHR和LHR的车道标线分别确定。线段扩展到矩形ROI的上下界。八个顶点的回报率决定的，所以这两个扭曲的梯形的形成和扩展线段位于中心；因此，检出率与以往的算法[ 10 ]三角形相比增加。

**C.车道标志跟踪:**在建立了ROI之后，在后续的帧中可以有效地检测和跟踪车道标记，因为在正常的驾驶条件下，下一帧中的车道标记将在ROI中找到。大多数噪声被排除在外，因为ROI只包括道路区域和面积小，因此计算量显著减少。在原有方案中，ROI具有固定边界的三角形，在修改后的ROI方案中，形状变为扭曲的梯形，边界是可变的。上部区域的短车道标记在修改后的ROI中更容易找到，而在旧的方案中往往会丢失，因为基于前面帧上的车道标记建立了ROI，并且当前帧中的车道标志的延伸上端不可能与三角形的顶点相重合。此外，如果在前一帧中没有检测到任何一方的车道标记，则修改后的ROI可以变得更宽。这使得车道标志在车道宽度改变或车辆水平超出允许范围时再次被检测到。

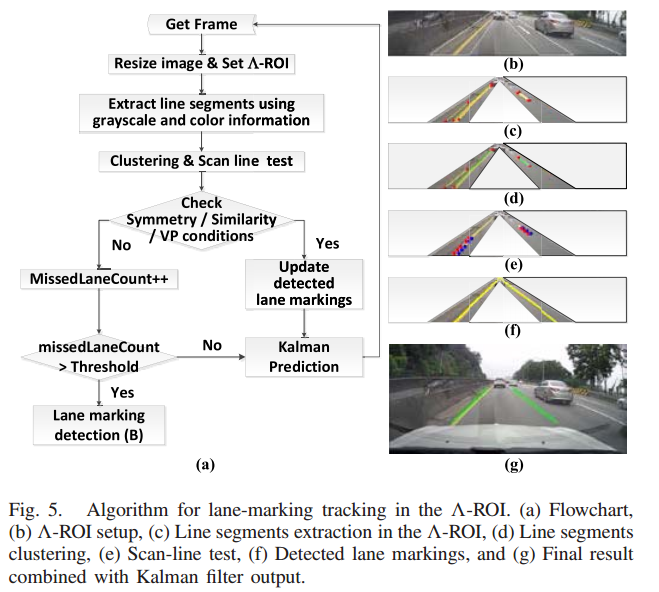


图5。车道标志跟踪算法。（a）流程图（b）- ROI设置，（c）线段提取- ROI（d）线段聚类，（e）扫描线测试，（f）检测车道标记，和（g）最后结果与卡尔曼滤波器输出相结合。

图5示出了跟踪ROI的车道标记的算法，该算法类似于车道标记检测阶段。首先，图像帧得到调整。然后将ROI配置并设置为活动区域。利用颜色信息的灰度图像，在ROI中提取线段。在聚类和扫描线测试之后，检查相似性、对称性和VP条件，调整线段的权重因子，对车道标志检测进行鲁棒性验证。如果一对以上的具有相同总权重因子的泳道标记候选满足条件，则选择具有较小距离的对，称为“内部对”。

与其他算法的比较，该算法与其他已有比较好的检测结果的算法相比，最近已经有报道。子（14）提出了一种光照不变的车道检测算法，该算法通过在单个帧中去除Canny边缘检测器和Hough变换获得的VP上面区域来配置ROI。它有时会导致一个不正确的副总裁时，投资回报率包括生存嘈杂的线路组件，虽然它在大多数情况下，它不删除的遮光罩和内部的车辆出现在图像的下部。所提出的自适应ROI算法在一定数量的帧上累积和跟踪VPS，从而消除噪声分量，并在图像的较低区域去除车辆中不必要的部分。儿子利用YUV颜色空间的分量Cb的黄色车道线检测。由强度和CB值组成的像素的二进制值是使用或操作组合的。在我们的实验中，黄色车道标志被侵蚀或被厚重的灰尘覆盖时，强度和黄色值都低于阈值。该算法在计算灰度和YUV颜色空间的| U-V |值并将其添加到增加值为黄色标志；同时，黄色的车道线检测即使被侵蚀或覆盖着厚重的灰尘。子采用连通分量聚类法15和最小二乘线拟合法。子算法中聚类的初始数据是二值像素，而基于边缘检测的算法是线段。虽然噪声对光照有很强的鲁棒性，但由于噪声线段的影响，子算法不能避免错误的检测。由于强车道标识验证过程，该算法对大多数噪声线段具有鲁棒性，并利用卡尔曼滤波改进了车道标志跟踪方案，可以恢复丢失的车道标志。

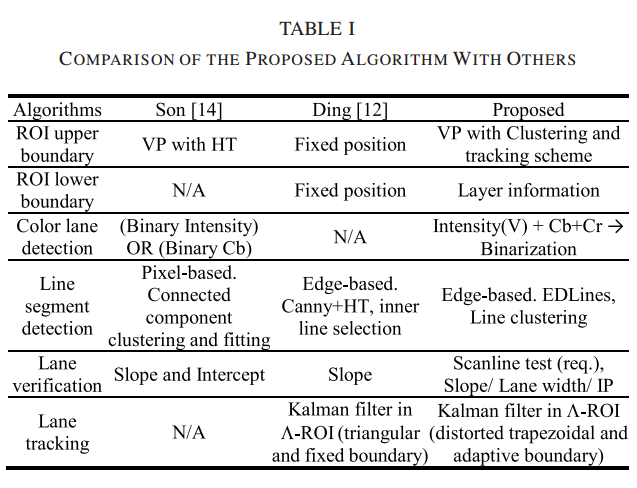
荣格（16）提出了一种基于时空图像的车道检测算法。虽然Jung的算法可以抑制噪声车道点附近的车道标线，它有几个潜在的问题。首先，当超过一定阈值的有噪声的车道点数量时，检测很可能失败。由于道路标线、其他车辆、照明、天气状况等原因，可以在远离车道标线的道路上获得有噪声的车道点。其次，Jung的车道检测算法会导致车道宽度的增加或减少，因为它试图找到垂直线，并依赖于道路宽度。此外，时空图像取决于车辆的速度，这将影响检测结果。

柳[ 17 ]提出了一种利用gradientenhanced灰度图像的车道线检测算法。虽然它是一种有效的算法来消除光照效果，但它也提高了道路或其他车辆上的噪声模式的边缘，在这种情况下是无效的。

上面提到的算法没有车道验证程序和车道跟踪计划，当道路和天气条件难以探测车道时，提高了检测率。该算法消除了道路的大部分噪声，车道验证过程不包括车道线附近的车道裂缝或新路面的边界。它也减少了计算量。车道跟踪方案可以检测车道即使车道makingsare完全侵蚀或不画一定数量的帧。表I总结了该算法与SON和以前的算法的比较结果。

1. **实验结果:**所提出的车道线检测算法是一个原型的嵌入式系统中使用C++和OpenCV库实现Ubuntu 14。其目的是在实时车辆中运行所提出的算法。目标嵌入式系统ARM Cortex-A9处理器的双运行在800 MHz。评价实施的软件的基础上提出的算法的性能，视频剪辑48（约92000帧）攻占美国和韩国使用仪表板摄像机收集；它们包括各种天气条件下，光照，和标志线[ 18 ]。

地面真实是通过绘制两条边界线沿着每一帧的车道标记的长边，并扩展到道路的ROI的上下边界，如果车道标记存在手动。考虑到量化误差，绘制了1个像素的边界线。虚拟边界线是根据前一帧的边界线生成的，如果车道标志不存在时，考虑到车辆的水平移动时，它们之间的距离增加了50%，如道路上的车道标识不在路上或道路正在施工中。将道路ROI分为3个垂直段，当检测到的车道标志位于中、下边界线之间时，确定车道检测是否正确。检测率是正确帧除以被评估帧总数的数目。



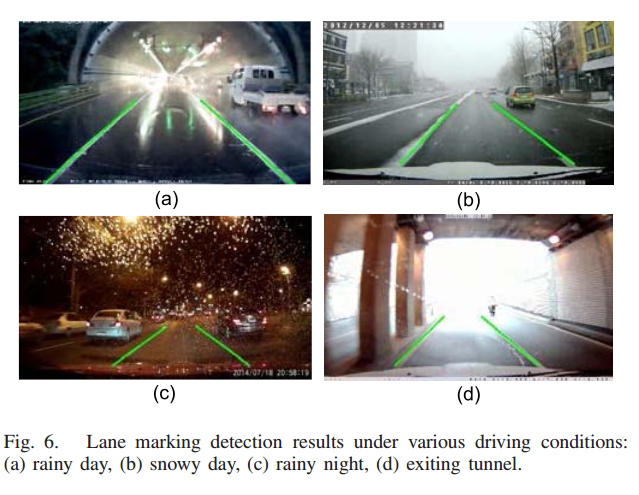


图6。各种行车条件下的车道标线检测结果：（一）雨天、（b）雪天、（c）雨夜、（d）出洞。

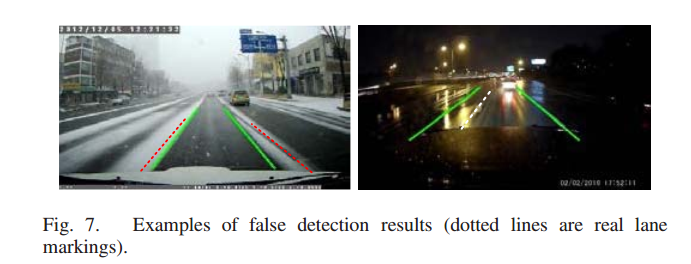
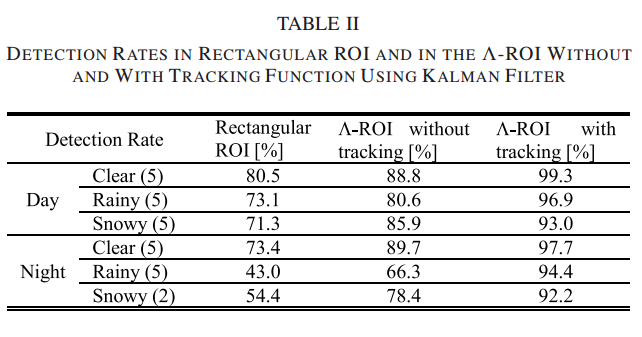
图6示出了用于各种天气和道路条件的实施软件和视频剪辑的成功的车道标记检测结果。无论是在图6（c）中所示的雨夜，挡风玻璃上的雨滴都能成功地进行车道标记检测，而图6（d）所示的隧道出口上的白色出口验证了该算法的鲁棒性。然而，该算法无法检测到车道标志，当他们完全覆盖积雪，当他们没有看到由于反射路灯在雨夜，如图7所示。

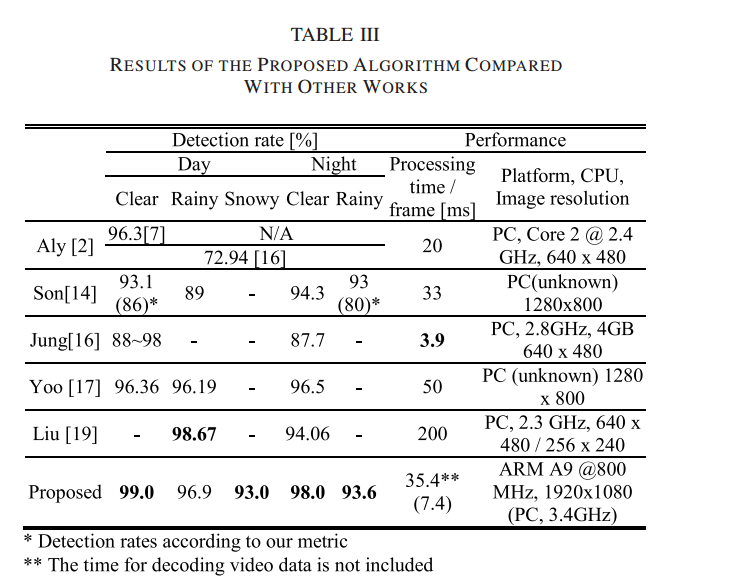
图7。假检测结果的例子（虚线是实线标记）。



表二显示了ROI和卡尔曼滤波在车道检测应用中的突出效果。这些行分别代表晴朗、雨天和雪天的白天和夜间的探测率。在天气条件标签旁边的括号中的数字代表了用于实验的视频数量。第一列表示矩形ROI中的检测率，第二列表示在ROI中没有使用卡尔曼滤波器的跟踪功能的列。表二显示，回报率是降低噪声和提高检测率非常有效；非ROI检测率提高了8.3%∼24%（平均14.7%）。第三列表示使用卡尔曼滤波器的ROI和跟踪功能的检出率。跟踪功能7.1%∼28.1%检测率提高（平均14.2%）。

即使道路被清扫后，积雪也常常覆盖着车道标线，积雪的白色会产生强烈的噪声成分。在夜间的道路上反射的街灯，使车道标记检测困难，因为车道标记甚至不可见的人的眼睛。表二显示，ROI和跟踪功能负责提高车道探测率的主要贡献。

表三：算法的结果与其他工程相比



所提出的算法的性能与以前的工作[ 2 ]，[ 14 ]，[ 16 ]，[ 17 ]，[ 19 ]比较。表三显示了检测率和每帧计算时间的比较结果。表三中的值是使用上述参考文献的数据复制或估计的，尽管视频数据集是不同的。阿离报道，在一个晴朗的日子[ 2 ]自己的数据库的检出率为96.3%；然而，阿离算法的检测率为72.94%根据柳的数据库包括视频剪辑，在晚上下雨天[ 17 ]。此外，计算检出率的不同度量的另一个问题是，儿子和柳的结果检测率为清晰、雨天和夜晚的条件下，该算法的结果。然而，根据我们的度量，儿子测量的检测率分别为晴天和雨夜的86%和80%，而根据儿子的度量，分别为93.1%和93%。这表明我们的度量比儿子更严格。表三表明，该算法的检测率优于其他工程，在这项工作中考虑到更多的道路条件比其他工程。特别是在其他作品中没有使用雪的视频剪辑。计算时间甚至优于其他算法。与嵌入式处理器相比，该算法比其他算法具有更快的处理速度。实际上，该算法可用于如嵌入式系统ADAS现实世界中的应用。

**四结论：**提出了一种高效、鲁棒的车道检测与跟踪算法。该算法由初始化、车道检测和车道跟踪组成。初始化使用VP找到一个矩形ROI，车道检测发现ROI，这是一个扭曲的梯形区域，围绕着车道标志。车道跟踪检测和跟踪车道标记的投资回报率。提出了引入颜色线索、线聚类、扫描线测试、车道验证和可变ROI等复杂的方法，该算法对含噪声成分的道路条件具有更好的性能。该算法使用48个视频剪辑进行了验证，这12个视频代表了道路和天气条件。计算时间满足嵌入式处理器在800 MHz运行时的实时操作。将该算法的检测率和计算时间与其他算法进行了比较，结果表明该算法优于其他算法。

REFERENCES  
[1] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, “Recent progress in  
road and lane detection: A survey,” Mach. Vis. Appl., vol. 25, no. 3,  
pp. 727–745, 2014.  
[2] M. Aly, “Real time detection of lane markers in urban streets,” in Proc.  
IEEE Intell. Vehicles Symp., Eindhoven, The Netherlands, Jun. 2008,  
pp. 7–12.  
[3] J. McCall and M. M. Trivedi, “Video-based lane estimation and tracking  
for driver assistance: Survey, system, and evaluation,” IEEE Trans. Intell.  
Transp. Syst., vol. 7, no. 1, pp. 20–37, Mar. 2006.  
[4] S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, and H. Chen, “A novel  
lane detection based on geometrical model and Gabor filter,” in Proc.  
IEEE Intell. Vehicles Symp. (IV), San Diego, CA, USA, Jun. 2010,  
pp. 59–64.

[5] Y. Otsuka, S. Muramatsu, H. Takenaga, Y. Kobayashi, and T. Monj,  
“Multitype lane markers recognition using local edge direction,” in  
*Proc. IEEE Intell. Vehicles Symp.*, vol. 2. Versailles, France, Jun. 2002,  
pp. 604–609.  
[6] P. M. Daigavane and P. R. Bajaj, “Road lane detection with improved  
canny edges using ant colony optimization,” in *Proc. ICETET*,  
Nov. 2010, pp. 76–80.  
[7] T.-Y. Sun, S.-J. Tsai, and V. Chan, “HSI color model based lanemarking detection,” in *Proc. IEEE Intell. Transp. Syst. Conf.*, Toronto,  
ON, Canada, Sep. 2006, pp. 1168–1172.  
[8] Z. Teng, J.-H. Kim, and D.-J. Kang, “Real-time lane detection by using  
multiple cues,” in *Proc. Int. Conf. Control Autom. Syst.*, Gyeonggi-do,  
South Korea, Oct. 2010, pp. 2334–2337.  
[9] L. N. P. Boggavarapu, R. S. Vaddi, H. D. Vankayalapati, and  
J. K. Munagala, “A robust multi color lane marking detection approach  
for Indian scenario,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 5,  
pp. 71–75, May 2011.  
[10] D. Ding, C. Lee, and K.-Y. Lee, “An adaptive road ROI determination algorithm for lane detection,” in *Proc. TENCON*, Xi’an, China,  
Oct. 2013, pp. 1–4.  
[11] C. Akinlar and C. Topal, “Edlines: Real-time line segment detection  
by Edge Drawing (ed),” in *Proc. 18th IEEE Int. Conf. Image Process.*,  
Brussels, Belgium, Sep. 2011, pp. 2837–2840.  
[12] D. Ding and C. Lee, “Efficient real-time lane detection algorithm using  
V-ROI,” *J. IKEEE*, vol. 16, no. 4, pp. 349–355, Dec. 2012.  
[13] Q. Lin, Y. Han, and H. Hahn, “Real-time lane departure detection based  
on extended edge-linking algorithm,” in *Proc. 2nd Int. Conf. Comput.  
Res. Develop.*, Kuala Lumpur, Malaysia, May 2010, pp. 725–730.  
[14] J. Son, H. Yoo, S. Kim, and K. Sohn, “Real-time illumination invariant  
lane detection for lane departure warning system,” *Expert Syst. Appl.*,  
vol. 42, pp. 1816–1824, Oct. 2014.  
[15] R. N. Hota, S. Syed, S. Bandyopadhyay, and P. Radhakrishna, “A simple  
and efficient lane detection using clustering and weighted regression,”  
presented at the COMAD, Mysore, India, Jun./Jul. 2009.  
[16] S. Jung, J. Youn, and S. Sull, “Efficient lane detection based on  
spatiotemporal images,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1,  
pp. 289–295, Jan. 2016.  
[17] H. Yoo, U. Yang, and K. Sohn, “Gradient-enhancing conversion for  
illumination-robust lane detection,” *IEEE Trans. Intell. Transp. Syst.*,  
vol. 14, no. 3, pp. 1083–1094, Sep. 2013.  
[18] *DSDLDE v.0.9: Video Clips for Lane Marking Detection*.  
[Online]. Available: https://drive.google.com/file/d/1315Ry7isciL-  
3nRvU5SCXM\_-4meR2MyI/view?usp=sharing  
[19] G. Liu, S. Li, and W. Liu, “Lane detection algorithm based on local  
feature extraction,” in *Proc. Chin. Autom. Congr.*, Changsha, China,  
Nov. 2013, pp. 59–64.