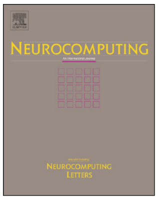
Neurocomputing 168 (2015) 336–347

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Video-based road detection via online structural learning

Yuan Yuan [a], Zhiyu Jiang [a,b], Qi Wang [c,*]

[a] Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, PR China
[b] University of the Chinese Academy of Sciences, 19A Yuquanlu, Beijing 100049, PR China
[c] School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, PR China

ARTICLE INFO

Article history:
Received 5 February 2015
Received in revised form
27 April 2015
Accepted 25 May 2015
Communicated by Xiaofei He
Available online 5 June 2015

Keywords:
Computer vision
Machine learning
Road detection
Structural SVM
Online updating
Road boundary

ABSTRACT

Video-based road detection is a crucial enabler for the successful development of driver assistant and robot navigation systems. But reliable detection is still on its infancy and deserves further research. In order to adapt to the situation consisting of environmental varieties, an online framework is proposed focusing on exploring the structure cue of the feature vectors. Through the structural support vector machine, the road boundary and non-boundary instances are firstly discriminated. Then they are utilized to fit a complete road boundary. After that, the road region is accordingly inferred and the obtained results are treated as ground truth to update the learned model. Three contributions are claimed in this work: online-learning updating, structural information consideration, and targeted sampling selection. The proposed method is finally evaluated on several challenging videos captured by ourselves. Qualitative and quantitative results show that it outperforms the other competitors.

1. Introduction

According to one recent report [1], road traffic injury remains an important health problem for the public. The total number of road traffic deaths keeps unacceptably high at 1.24 million per year, while the primary cause is unacceptably due to the driver's inattention and tiredness. To alleviate this situation, *Driver Assistance System* (DAS) [2–4] is developed and equipped, with the hope that it can serve as an autonomous reminder and guidance for the drivers. Among the various techniques enabling the DAS, road detection is the fundamental one, because it is the first step for a vehicle to become moveable and many other intelligent maneuvers are based on it. For example, *Lane Departure Warning* (LDW) [5,6], Lane Centering [7], and even full autonomous driving [8] rely on the results of road detection. Moreover, it can provide a significant contextual cue for target detection (e.g, vehicle or pedestrian) [9–12] and act as the prerequisite for robot navigation in an outdoor environment, which is widely researched in artificial intelligence and computer vision.

Because of its practical and theoretical importances, road detection has been thoroughly investigated in recent years. According to the types of sensing modalities used for this purpose, existing methods can be categorized into active sensor based and passive sensor based. For the *active sensor based* methods, the sensors project certain kinds of radiative lights and measure the reflection from its projection. Typical examples include *Light Detection And Ranging* (LIDAR) and *Radio Detection And Ranging* (RADAR). Several active sensors have been widely used for road understanding and great progress has been made since the DARPA Grand Challenge and Urban Challenge [13].

However, due to the restriction of limited perceptual range by the active sensors, and the risk of inter-vehicle inference or pollution to the environment, the *passive sensor based* methods have a tendency of dominating the trend because of their noninvasive characteristic. To be specific, the passive sensors obtain useful information from the environment by capturing the reflection of sun light or other artificial lights. This kind of method can provide intuitive understanding of all the surrounding environment and deliver more meaningful cues, which are crucial for the development of future intelligent transportation systems in mixed traffic conditions [14]. As for the sensor, video cameras that provide the visual data are the most frequent choice. Therefore, the term “video-based” is interchangeably used with “passive sensor based” for simplicity in the following sections.

In this paper, we address the problem of video-based road detection utilizing an online strategy. The major focus is on

[*] Corresponding author.
E-mail address: crabwq@nwpu.edu.cn (Q. Wang).

http://dx.doi.org/10.1016/j.neucom.2015.05.092
0925-2312/© 2015 Elsevier B.V. All rights reserved.

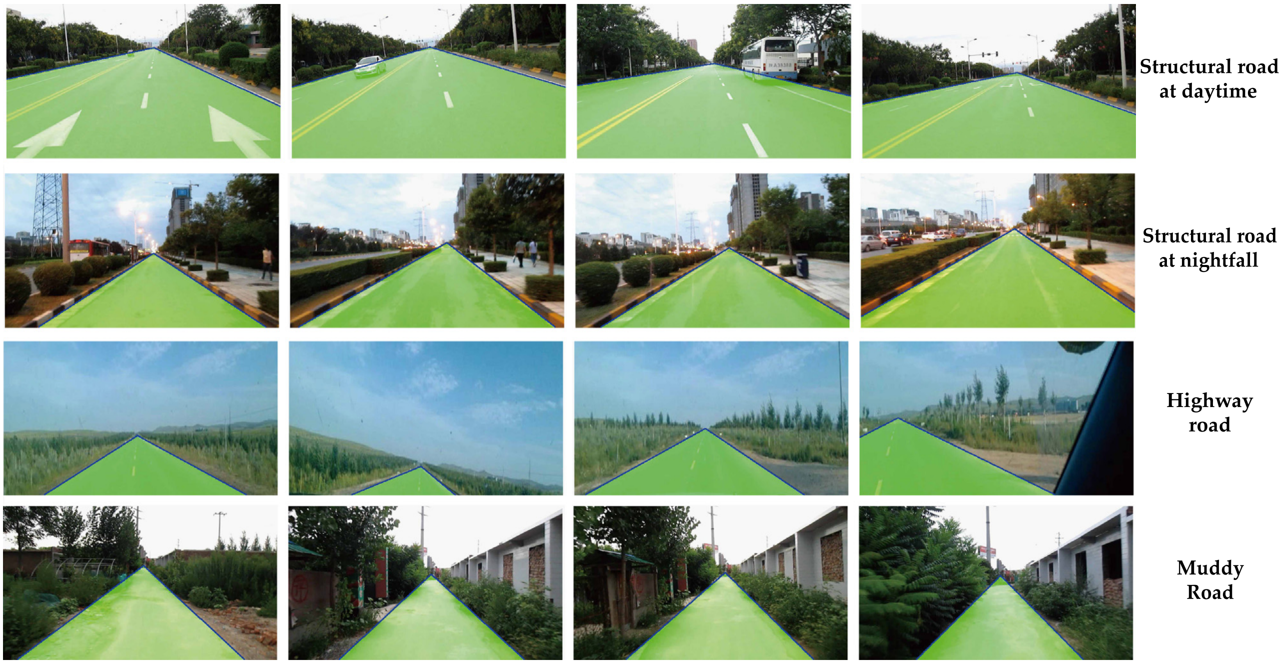


Fig. 1. Typical road detection results of the proposed method. From top to bottom, each row represents the detection results of a specific kind of scenery.

exploring the structural information of the input data through the structural SVM (SSVM). At the same time, the learned model is online updated to adapt to the changing environment. Fig. 1 shows some typical detection examples using the proposed method.

1.1. Related works

Since the presented work belongs to the passive sensor based type, we only review the video-based methods. With respect to the different emphasis on the prior knowledge, road detection can be divided into three groups: *model-based*, *feature-based*, and *learning-based*.

(1) *Model-based method* tends to have an assumption of road shape, which is actually treated as road model. Then the aim is to find the fittest parameters under the model assumption. Several strategies of model fitting [15–19] are used to get the road model. Oniga et al. [15] fitted a quadratic road model by RANSAC approach and the fitting result was refined by a region growing-like process so as to determine the road surface. Sappa et al. [16] proposed Least Square Estimation (LSE) based approach to fit a model for the road surface. Fardi et al. [17] utilized Hough domain to determine the road borders after using the Gaussian pyramid technique to model the scale information. Borkar et al. [18] employed RANSAC to eliminate outliers caused by noise and artifacts in the road and Kalman filter was finally used to smooth the road boundaries. Sawano and Okada [19] utilized an internal energy based on the tendency of a control point resisting changes in its state of motion in an image space, to represent the road model. Although *Model-based methods* can accurately determine the road region given a proper road model, it may be invalid to face the situations where road shapes change as the vehicle is moving. Therefore, it is difficult to find an appropriate model for unstructured roads with inconstant conditions.

(2) *Feature-based method* relies upon the extraction of image features to detect road boundaries and road region. The features such as color, gradient and texture are commonly used to measure local neighborhoods and a likelihood function is formulated by feature clustering [20], threshold segmentation [21] or region growing approach [22] to obtain the road region. For example, He et al. [23] assumed that the color components of road surfaces obey the Gaussian distribution and the road areas were detected based on the full color features. Sotelo et al. [24] utilized the Hue-Saturation-Intensity (HSI) color features for segmentation to model the road pattern. Alvares et al. [25] employed an illuminant-invariant, which was converted from the RGB space, as the feature space to accomplish the road detection task. The main advantages of the *feature-based method* are that it is insensitive to the shape of roads and little previous knowledge is needed. But it is sensitive to shadows and other illumination changes.

(3) *Learning-based method* [25–27] generally makes use of a trained neural network or classifier to distinguish between the road region and non-road region. Such methods are independent of special road markings and are capable of dealing with non-homogeneous road appearance, if the characteristics of road or non-road regions are properly represented by the feature space. Alvarez and Lopez [25] introduced a shadow-invariant feature space and it was used along with a likelihood-based classifier which was online learned to achieve road segmentation. Son et al. [27] constructed a probabilistic road model by supervised training and a posteriori probability based on visual information was then utilized to extract the road region. For *learning-based method*, although less prior knowledge is needed, it heavily relies on the training sets and training strategies. But unfortunately, most of the classifier and neural network are trained once, unable to adapt to the varieties of the environment.

Apart from the three types, most road detection problems can be successfully interpreted using a variant of the three above approaches or a combination of them. The proposed method in this work belongs to the learning based prototype, while taking advantages of advanced features and road boundary fitting.

1.2. Proposed framework

Though many works have been proposed, most of them are based on the assumption that the road area is consistent in intensity or color. However, in real-life environments, this assumption might fail because the intensity often varies a lot as the vehicle or robot is moving. Moreover, the shadows and occlusions

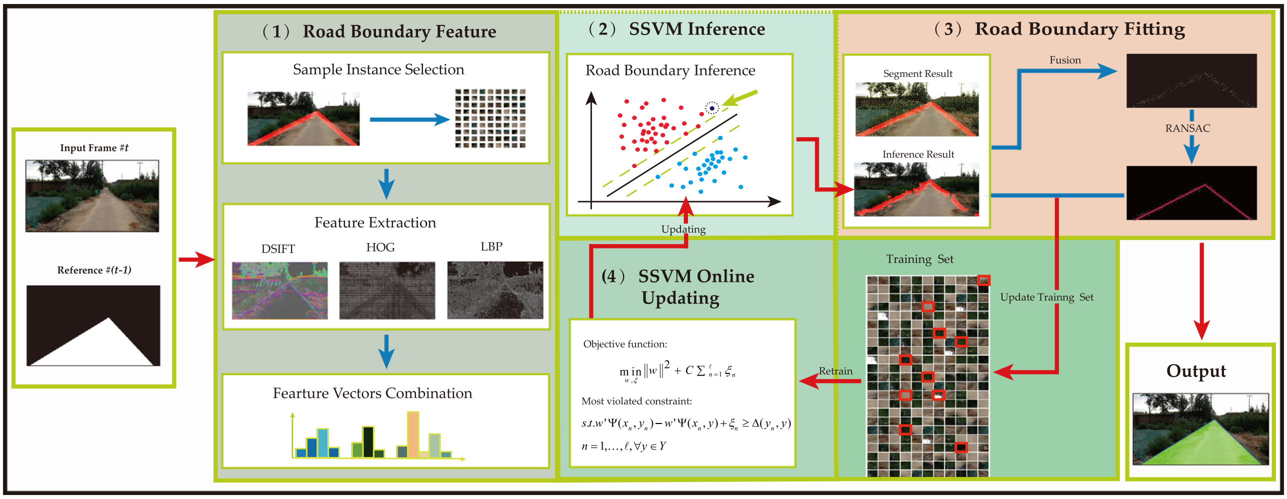

Input Frame #t
Reference #(t-1)
(1) Road Boundary Feature
Sample Instance Selection
Feature Extraction
DSIFT
HOG
LBP
Feartture Vectors Combination
(2) SSVM Inference
Road Boundary Inference
Updating
(3) Road Boundary Fitting
Segment Result
Fusion
RANSAC
Inference Result
(4) SSVM Online Updating
Objective function:
$\min_{w,\xi} \|w\|^2 + C\sum_{n=1}^{\ell} \xi_n$
Most violated constraint:
$s.t. w'\Psi(x_n, y_n) - w'\Psi(x_n, y) + \xi_n \geq \Delta(y_n, y)$
$n = 1, \ldots, \ell, \forall y \in Y$
Retrain
Training Set
Update Trainng Set
Output

Fig. 2. Road detection pipeline. For an input frame at time t, the task is to detect the road area with reference to previously obtained detection result at time $t-1$. The first step is to select a set of sample instances and express them with a fusion of three features. Then a SSVM classifier is utilized to distinguish the road boundary and non-boundary instances. After that, a line fitting process is applied to get a more complete and reliable boundary, abandoning those outliers. At the same time, the road area can be inferred from the obtained boundary. Finally, to keep the model adaptable to the changing environment, an updating strategy is enforced by emphasizing more on the misclassified instances. The updated model will be used in the next frame.

would also influence the detection results. In this paper, we focus on the drivable road detection, aiming at inferring the road region in a video collected by a camera mounted ahead of a vehicle or robot. Particularly, the road region is inferred from the road boundary, which is not restricted to only structural roads with distinguished lanes or curbs. Fig. 2 illustrates the pipeline of the proposed method, which is named as *Road Detection via Online Learning* (RDOL). The four main components are introduced as follows.

(1) *Road feature extraction*: In this work, we tactfully transform the road detection problem into detecting the road boundary. The ultimate road region can be constrained and inferred from the detected boundary. The reason for detecting the road boundary instead of road region can be explained as follows. Firstly, compared with road boundary, more pixels are involved for the road region and it is more likely to be influenced by the intensity change, shadow and lighting condition. These phenomena will result in large intra-class differences and degenerate the model's discriminative ability. Secondly, road boundary always coincides with the image edge which is less ambiguous compared with the region based method. As for the determination of road boundary, the feature selection is a critical factor. In this work, only neighbor information of road boundary is considered. The involved features include local gradient and texture, for the reason that they are significantly manifest for road boundary. Therefore, DSIFT (Dense SIFT), HOG (Histogram of Oriented Gradient) and LBP (Local Binary Pattern) are used for their robustness to intensity change and shadow.

(2) *SSVM inference*: After feature extraction, a classifier is employed to determine the existence of road boundary in the examined frame. In this step, we assume that the boundary in the previous frame has been detected and a classifier for boundary/non-boundary has been learned. Both of them are available for reference. Then a portion of sample instances are selected from the current frame as the candidates for further verification. These instances are purposefully chosen from the locations near the previous boundary because the adjacent frames are prone to have little difference in boundary location. After that, the sample instances are tested by the previously learned structure SVM classifier, with an output of binary labels. The classifier considers the structure cue among input data and is robust for the adaptive environment.

(3) *Road boundary fitting*: After classification, the detected road boundary consists of sparse points and not reliable enough. Accordingly, a segmentation result is used to improve it. Our assumption is that the region edges of obtained superpixels are more likely to share the same positions with the detected boundary pixels. Those coincident pixels are treated as the reliable ones to support further boundary fitting. Considering the fact that straight road is the most common case in daily life, especially when the camera is close to the ground, straight line fitting is investigated as an example in our framework. The other situations are similar to it.

(4) *SSVM online updating*: There is a strong possibility that the road appearance varies as the vehicle or robot is moving. Consequently, an online updating strategy is critical. When finishing the boundary detection procedure, the road region is inferred from the extracted boundary and it is treated as ground truth. At the same time, the structural SVM classifier is updated with the misclassified samples from the examined frame. The retrained classifier claims to be adaptive to the changed scene and will be utilized in the next frame.

1.3. Contributions

Although many road detection methods are proposed in the literature, the presented method in this paper still has its advantages. The main contributions of this research are listed as follows.

(1) *Online-learning framework*: Traditional methods use a fixed training set to generate the classifier. However, it is known to us that the road features, such as intensity, color and gradient, may change dramatically in a variety of environmental conditions. An adaptive strategy might be necessary for a robust performance. In view of this point, this work emphasizes on the online-learning ability of the designed detector and the updated model maintains capable of tackling the novel environmental changes.

(2) *Structural information considered*: For traditional classification problem, the structural information among the training instances is not considered. In fact, different classes may have different underlying data structures. It requires the classifier to adjust the discriminative hyperplane to fit for the data structure. As for the context of road detection, we think that the boundary and non-boundary samples have their respective characteristics and data distribution. Therefore, structural SVM is introduced as

the classifier for its structured learning ability and the online implementation is also used to accelerate the computation speed.

(3) *Targeted sampling selection*: Generally, any of the positive and negative samples can be selected for the training purpose. In this work, only specific neighbor pixels around the road boundary are considered. We think the regions far from the boundary will cause the degeneration of the classifier because of the lighting differences and shadows. As for the update of the model, only the misclassified instances are involved in the retraining procedure because they are more informative. By meaningfully selecting the training samples, the classifier would become more robust and discriminative.

Moreover, these three contributions are generated into a unified framework for the first time. Online learning strategy makes the proposed method adapt to varying road scenes. Structural information in the feature space helps to refine the road detection results which can improve the performance of online learning in return. The target sampling selection can also avoid the degeneration of the model which online learning suffers. To sum up, all these contributions are considered based on the road detection problem and the detection results would be reliable and robust.

The remainder of this paper is organized as follows. In Section 2, the proposed method is described in detail, including feature extraction, classification framework, boundary fitting and model updating. In Section 3, experimental results are presented, with a comprehensive qualitative and quantitative comparison and analysis. Finally, conclusion is made in Section 4.

2. Road detection

The aim of this work is to detect the road area in an input video, which is collected from a vehicle-mounted or robot-mounted camera. Since region-based road detection is sensitive to intensity change, shadow and lightening condition, we transform the road detection problem into detecting the road boundary. Then the road region is accordingly inferred from the obtained boundary. Traditional road boundary detection methods mainly focus on the intensity space and utilize fitting strategies to estimate the candidate boundaries. Our method puts the boundary fitting on the more robust feature space and regards the road boundary detection as a binary classification problem.

Since the proposed method is based on adaptive learning, a manually labeled ground truth is needed in the first frame. Then the model is updated at each frame. In the following part, we will introduce the proposed method step by step.

2.1. Road boundary feature extraction

In this work, the image patch centered at a sampling pixel is defined as an instance. To appropriately represent the instances and distinguish the positives from the negatives, transforming the image from intensity space into feature space is a reasonably better choice. This is because the intensity could change a lot when the vehicle or robot is moving and its absolute value has little meaning to a region especially when it passes through the edge. Based on these characteristics, gradient and texture feature are considered in the proposed method and three different feature spaces are used, including Dense SIFT (*DSIFT*), Histogram of Oriented Gradients (*HOG*) and Local Binary Pattern (*LBP*).

2.1.1. DSIFT feature

SIFT feature [28] has been widely used in computer vision, for it is robust and invariant to scale, noise and illumination. DSIFT is obtained by computing the SIFT descriptor of the same scale over dense grids in the image domain. In our experiment, the sampled instances are $N \times N$ image patches. The feature vector of each instance can be achieved with a 4×4 array of histograms in which it has 8 orientation bins. Therefore, the dimension of DSIFT descriptor is $4 \times 4 \times 8 = 128$. The obtained DSIFT feature is denoted as $\{\mathbf{d}_i \,|\, i = 1, 2, \ldots \ell\}$, where ℓ represents the number of sampling instances in the current frame and $\mathbf{d}_i$ is the ith instance descriptor.

2.1.2. HOG feature

HOG feature [29] was first proposed for the problem of human detection. Ever since then, numerous experiments have proved the effectiveness of HOG, because it is invariant to changes in lighting, small deformations, etc. In this work, we also take HOG into the boundary detection task. Similar to SIFT, the original HOG computes a histogram of gradient orientations in a local block. In our experiments, sampling instances are considered as a local block and each block would generate one column feature vector. It is represented as $\{\mathbf{h}_i \,|\, i = 1, 2, \ldots \ell\}$ where $\mathbf{h}_i$ is the 31-dimensional descriptor. The dimension of the HOG descriptor can be adjusted by changing the sampling distance of the histogram.

2.1.3. LBP feature

LBP [30] is a powerful descriptive method for texture information. The original LBP operator labels the image by thresholding the $N \times N$ neighbors of each pixel compared with the center pixel in intensity space and considering the result as a binary number. Then texture descriptor can be generated by converting the binary number to decimal number or counting the histogram of the labels. One extension to the original LBP operator is the introduction of uniform patterns [31] which is used in the proposed method and a new pattern can be categorized into either one of the uniform patterns or non-uniform pattern. The LBP feature vector of the ith patch is denoted as $\{\mathbf{l}_i \,|\, i = 1, 2, \ldots \ell\}$ where $\mathbf{l}_i$ is the 10-dimensional descriptor. The dimension of the LBP descriptor can also be adjusted by sampling distance of the histogram.

After the extraction of the features mentioned above, the feature vector $\mathbf{x}_i$ of the ith instance is obtained by concatenating each kind of feature vector into a column vector, and written as

$$\mathbf{x}_i = [\mathbf{d}_i; \mathbf{h}_i; \mathbf{l}_i], \quad i = 1, 2, \ldots \ell, \tag{1}$$

where $\mathbf{d}_i$ represents DSIFT feature vector of the ith patch, $\mathbf{h}_i$ is the HOG feature vector and $\mathbf{l}_i$ denotes the LBP feature vector.

2.2. Online structural SVM

With the obtained feature vectors, a SSVM classifier is adopted to make a binary decision of boundary/non-boundary. For the fact that the instances belonging to the same class may have the same data structure distribution in the feature space, this structure constraint can obviously eliminate the outliers. The classifier is learned in the first frame and updated in every following frame.

2.2.1. Training set generation

Generally, the ground truth of the first frame can be generated by region growing strategy assuming the middle-bottom pixels belong to road region. This assumption is always effective in real situations. On the other hand, it can also be manually labeled if the assumption is incorrect. Since we have a ground truth of road boundary in the first frame, the positive and negative training instances can be identified. The positive instances are uniformly sampled from the road boundary. The negative instances can be sampled from all the other non-boundary regions. However, the training is more effective if the negative instances are chosen near the road boundary. This is because the surrounding environment

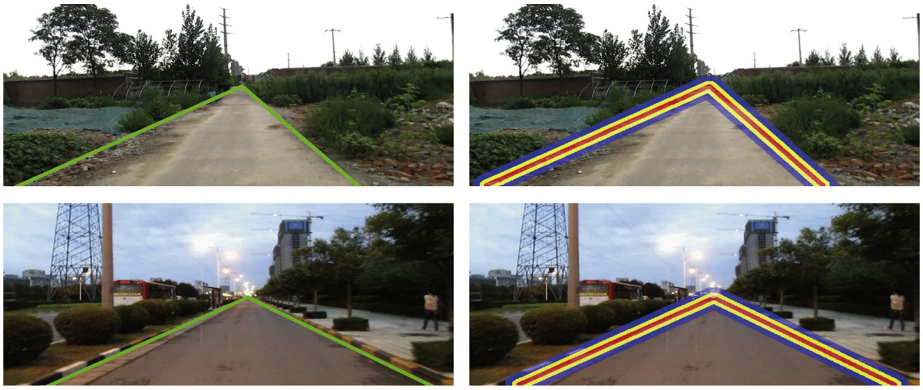


Fig. 3. Illustration of training sample selection. For the left images, the green lines indicate the ground truth road boundary. For the right images, positive instances can be randomly selected from the red region and negative ones from the blue region. The yellow region is served as a transition area. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

varies a lot. Negative instances far from the boundary would make the classifier more ambiguous and degenerate its accuracy. Therefore, negative instances are merely sampled from the neighbor of road boundary. This targeted sampling can ensure that the classifier's discrimination is strengthened. In addition, the computational complexity is reduced. The training set is denoted as

$$\mathbf{T} = \{(\mathbf{x}_i, y_i) \big| (\mathbf{x}_i, y_i) \in \mathcal{X} \times \mathcal{Y}, \quad i = 1, \ldots, \ell\}, \tag{2}$$

where $\mathbf{x}_i \in \mathcal{X}$ represents the ith feature vector and $y_i \in \mathcal{Y}$ is its label, ℓ is the number of sampled instances for the classifier training. Fig. 3 illustrates the sampling procedure of positive and negative instances.

2.2.2. *Road boundary inference*

For the training set $\mathcal{X}$ composed of a number of instances $\{\mathbf{x}_i\}_{i=1}^{\ell}$, the goal of road boundary inference is to find the related label $\{y_i\}_{i=1}^{\ell} \in \{-1, +1\}$ of each instance. For simplicity, $\mathbf{x}_i$ is replaced by $\mathbf{x}$. Instances considered either as road boundary or not can been taken as an "one against all" classification problem. And the problem can be cast as learning a prediction function $f : \mathcal{X}$ to $\mathcal{Y}$ mapping the feature instance $\mathbf{x}$ to the binary classification label y. For better considering the structural information between instances, structural SVM is utilized to learn this prediction function. Dissimilar to traditional classifiers such as SVM, SSVM considers the separability between classes and the compactness within classes simultaneously. It directly introduces the data distributions of classes into the optimization function. By this means, the underlying data structure is emphasized in the classification procedure.

As for the implementation, f is actually defined to be a score function $F : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$. Generally, F is chosen as a linear combination of $\mathbf{w}$ and $\Psi(\mathbf{x}, y)$, written as

$$F(\mathbf{x}, y; \mathbf{w}) = \langle \mathbf{w}, \Psi(\mathbf{x}, y) \rangle, \tag{3}$$

where $\mathbf{w}$ is the weight parameter and $\Psi(\mathbf{x}, y)$ is the joint feature transformation. Once their values are set, the corresponding label of an input feature vector $\mathbf{x}$ can be inferred by maximizing the output of F, which means

$$f(\mathbf{x}; \mathbf{w}) = \underset{y}{\operatorname{argmax}} F(\mathbf{x}, y; \mathbf{w}). \tag{4}$$

In this process, $\Psi(\mathbf{x}, y)$ is denoted as a kernel function that maps the feature space $\mathbf{x}$ to a higher dimension space in order to further improve the classification performance [32]. Meanwhile, the distribution of the instances in the same class, which represents its structural information, is also reflected in this function. For the instances in the same class, the same mapping strategy according to $\Psi(\mathbf{x}, y)$ is utilized to project the instances into the same higher dimensional space. Different classes would be projected into different higher dimensions.

As for the estimation of $\mathbf{w}$, it depends on a set of training instances $\mathbf{T}$ which are obtained from the training set generation mentioned above. The structured SVM learns the parameter $\mathbf{w}$ through the minimization of a constrained quadratic optimization problem utilizing the risk minimization and margin maximization strategies:

$$\begin{aligned} \min_{\mathbf{w}, \xi} \quad & \|\mathbf{w}\|^2 + C \sum_{i=1}^{\ell} \xi_i, \\ s.t. \quad & \mathbf{w}'\Psi(\mathbf{x}_i, y_i) - \mathbf{w}'\Psi(\mathbf{x}_i, y) + \xi_i \geq \Delta(y_i, y), \\ & i = 1, \ldots, \ell, \forall y \in \mathcal{Y}, \end{aligned} \tag{5}$$

where C is a balance parameter. The function $\Delta : \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{R}_+$ measures a distance in label space. For binary labels, writes

$$\Delta(y_i, y) = (1 - y_i y)/2. \tag{6}$$

2.2.3. *SSVM online implementation*

There are a number of constraints in (5) with respect to the dimensionality of y. Therefore, the computational efficiency is a critical problem. In order to speed up the calculation, an online learning implementation is utilized to speed up the training procedure.

Algorithm 1. SSVM online implementation.

```
Input: Training set feature $\mathbf{X}$, corresponding label $Y$, threshold $\gamma$.
Output: Classification hyperplane $\mathbf{w}$.
Initialization: Randomly initialize $\mathbf{w}$.
Set $flag = 1$, accumulated cost $C = 0$, constraint set $T = \varnothing$;
while $flag == 1$ do
    Set $flag = 0$;
    for each training instance do
        Find the most violated constraint under the current $\mathbf{w}$;
        Add the most violated constraint to the constraint set $T$;
        Calculate the accumulated cost $C$;
        if $C > \gamma$ then
            Solve Eq. (5) under the constraint set $T$ via cutting plane algorithm;
            Update $\mathbf{w}$;
            Reset accumulated cost $C$;
            Set $flag = 1$;
        end if
    end for
end while
```

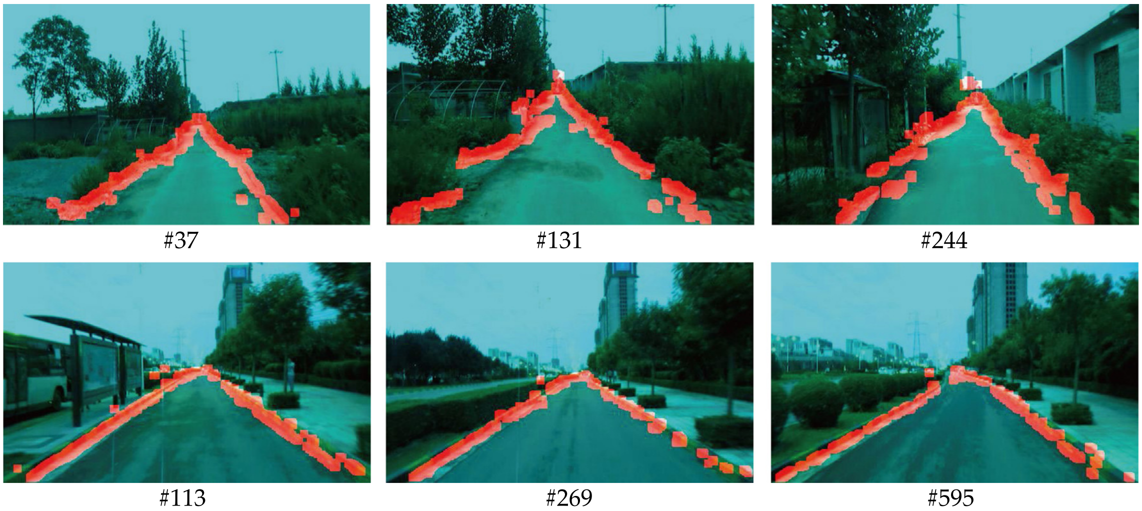

#37
#131
#244
#113
#269
#595

Fig. 4. Typical road boundary inference results. The detected boundary instances are labeled in red. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

The details of SSVM online updating is shown in Algorithm 1. For each training instance, the most violated constraint, which have the maximum error, is incrementally added to a constraint set which are combined by linear inequalities considering the structural information between instances of the same class. Then the cutting plane algorithm [33] would iteratively refine before analysing the constructed constraint set to solve the minimization problem. In this way, the optimization can be efficiently realized.

Fig. 4 shows some typical inference results of SSVM classifier which are randomly selected from the testing data.

2.3. Road boundary fitting

The inferred boundary instances deliver a certain level of true information, but they are not continuously connected from place to place. In order to make the detected road boundary more reliable, it is necessary to consider other constraints which would improve the robustness of the boundary inference result. For this purpose, edge and road model are the two constraints considered here.

As for the edge inspired constraint, our assumption is that the inferred road boundary instances will mostly exist in the same place where they can be detected by other edge detection methods. For commonly used edge detection operators, such as Canny, Sobel and Prewitt, they can only estimate the local edge which is sensitive to the noise in the pixel space. Though noise removal preprocessing can improve the performance, it is essential to consider more information in a bigger neighborhood. Fortunately, image segmentation can solve this problem well which integrates region-based cue and is robust to local noise. The desired edges can be easily obtained from the superpixels after segmentation [34]. In this case, even if the inferred boundary is unreliable (especially in faint object boundaries and cluttered background), the added edge cue from segmentation can enforce the credibility. Those instances with road labels from both sources are treated as the final detected road boundary instances. In our framework, SLIC superpixel method [35] is adopted. It clusters pixels in the combined five-dimensioned color space and it can generate compact, nearly uniform superpixels efficiently.

As for the road model, it is mainly employed to complete the disjoined boundary sections. In this work, the focus is primarily on the straight road detection. This is because the straight type is mostly seen in our daily life. Besides, the linear perspective phenomenon makes the curved line more "straight" especially when the camera is close to the ground plane. Thus the task accordingly becomes to obtain a set of "inliers" satisfying the straight line model. There are many ways for line fitting, but the robust estimation by random sampling, especially RANdom SAmple Consensus technique (RANSAC), has been proven to be the most promising one. In our framework, the detected instances are fitted through RANSAC method [36,37] which chooses the solution that maximizes the likelihood rather than just the number of inliers. After the road boundary fitting procedure, the road region can be easily identified as the region enclosed by the two straight lines.

The edge and road model constraints generally make the road boundary detection more reliable. The SSVM classified result may not be reliable enough, i.e. some non-road boundary parts may be regarded as road boundary. When the non-road boundary parts are incorrectly classified as road boundary, the edge constraint obtained by superpixel segmentation would eliminate these parts. Furthermore, the road model constraint would restrict the fitted curved lines of road boundary. To sum up, these two constraints can make final road boundary more reliable.

2.4. Online classifier update

Consider the situation that the road appearance, such as lighting, gradient, texture, etc., changes a lot as the vehicle or robot is moving. As a result, the classifier performance would degenerate without adaptation to the new road boundary condition. Given this fact, an online updating procedure is desired, which would make the classifier have more adaptability to the dynamic scene. In our framework, after straightline fitting, misclassified instances can be obviously distinguished by treating the obtained boundary as a ground truth label. In other words, for a positive instance, we consider it misclassified only if it is labeled negative by the classifier. Similarly, for a negative instance not belonging to the road boundary, we consider it misclassified only if it is labeled positive.

Just as the Adaboost updating procedure [38], only the misclassified instances are used to update the model. Let $D^p = \{(\mathbf{x}_i^p, y_i^p), i = 1, \ldots, n^p\}$ be the collection of n^p misclassified positive instances at the tth frame, and $D^n = \{(\mathbf{x}_i^n, y_i^n), i = 1, \ldots, n^n\}$ the collection of n^n misclassified negative instances. Considering the problem of sampling balance, we use a parameter τ to adjust the ratio between the number of positive instances and the number of negative instances, which can be interpreted as the knowledge of prior probability. Then the sampling number of positive is

$$k = \min\{n^p, n^n/\tau\}. \tag{7}$$

The updating training set would be

$$D = \left\{ D_i^p, D_j^n \middle| i \in rand(n^p, k), j \in rand(n^n, \tau k) \right\}, \tag{8}$$

where $rand(n, k)$ means randomly select k numbers from $\{1, \ldots, n\}$. These misclassified instances can also be regarded as "support vectors" which means the instances can make the classifier more robust.

After the training set updating by solving Eq. (5), the classifier would be updated and the renewed classifier would be used for the inference in the next frame.

3. Experiments and analyses

In this section, experiments are conducted to verify the effectiveness of the proposed method. We first introduce the data set constructed by ourselves. Then the experimental settings are detailed and the parameter selection is conducted. After that, experimental analysis and discussion are finally presented.

3.1. Data set

For validation of the proposed method, a data set of four kinds of road videos has been collected. The videos are all RGB image sequences and the acquisition rate is 25 fps. In our framework, the size of video frames is normalized into 300 × 500. In order to validate the robustness and effectiveness objectively, we have manually labeled almost 7000 frames of all the videos frames and the labeled results are served as the ground truth. The detailed attributes of these videos are listed in Table 1.

- *Structured road in daytime* ("*srd*" *sequence*): The first video contains structured road which was captured in daytime. This kind of road often has clear boundaries and they can be easily distinguished from other non-boundary regions. Several *T*-intersection road conditions are also included in the video.
- *Structured road in nightfall* ("*srn*" *sequence*): The second one is structured road which was taken at nightfall. Compared to the structural road in daytime, this video has lower intensity value and the neon lights make the road region much more similar to the surroundings in color appearance.
- *Unstructured highway road* ("*uhr*" *sequence*): The third video is on highway road. In this case, the road markings and pavements are not as good as the structured road. The boundaries of the road region is not clearly differentiated from the surroundings. The camera's jitter also makes the detection more difficult.
- *Unstructured muddy road* ("*umr*" *sequence*): The fourth one is muddy road. In this condition, the road surface is uneven and no obvious road boundaries exist. This makes the problem much more challenging.

Table 1
The data set description.

Video type	Number of frames	Attributes
"*srd*" sequence	4000	**T-intersection road**
"*srn*" sequence	977	**Low intensity, street lamp lighting reflection**
"*uhr*" sequence	1238	**Large pitch and yaw change, wide baseline**
"*umr*" sequence	580	**No clear boundary, variational road boundary**

3.2. Experimental settings

Before detailedly analyzing the performance of the proposed method in this paper, the competitors and evaluation criteria will be introduced in the following part.

1 *Competitors*: To verify the effectiveness of the proposed method, we compare it with state of the art. In this work, two competitors are employed, which are SVM-based and vanishing point-based methods. Since the SVM classifier has been widely used in computer vision applications, we firstly adopt it to classify the road boundary and then the following processing keeps the same to the proposed method. On the other hand, this setting can test the usefulness of structural information in the proposed framework. As for the vanishing-point-based method [39], the dominant texture orientation at each pixel is computed in a novel adaptive soft voting scheme using confidence-weighted Gabor filters. Then a vanishing-point constrained edge detection technique is used to detect road boundaries. Although no prior knowledge is needed, this method relies on time-consuming filters and the inference is sensitive to other straight edges in the scene. The road region is finally inferred from the boundary.

2 *Measurement for road region*: For the evaluation of the detected road region, we mainly focus on the pixel-wise metrics. Accuracy, precision, and recall are used in this work, which measure different aspects of the road detection results. The three metrics are defined based on the confusion matrix as described in [40,41]. To be specific,

$$\text{accuracy} = \frac{TP+TN}{TP+FP+TN+FN},$$
$$\text{precision} = \frac{TP}{TP+FP},$$
$$\text{recall} = \frac{TP}{TP+FN}. \tag{9}$$

But unfortunately, these metrics are sometimes contradictory. In order to get a unified evaluation, *F*-measure is usually used as a compromise. It is defined as

$$F-measure = \frac{precision \times recall}{(1-\alpha) \times precision + \alpha \times recall}, \tag{10}$$

where α is an adjustable parameter which is set to 0.5 in our experiments.

3 *Measurement for vanishing point*: In our framework, after the road boundary fitting procedure, vanishing-point can be obviously inferred from the crossover of the two straight lines. Based on the manually labeled ground truth, the Euclidean distance in pixels is used to show the quality of the vanishing-point estimation. The measurement can be calculated as follows:

$$\text{Distance} = \| P - P_0 \|, \tag{11}$$

where P is the estimated vanishing-point location and P_0 is the labeled vanishing-point location. All the input frames are normalized into 300 × 500 so as to make the proposed method adapt to more environmental conditions without any parameter changing.

3.3. Parameter selection

There are several critical parameters to be set in the experiments. The first one is the region size $N \times N$ of each sampled instance. In our method, the feature vector of the examined instance is expressed by the information contained around the neighborhood. Its size will influence the expression of the feature vector, and accordingly, the results of boundary inference. In order to select a suitable *N*, we conduct comparisons under different settings. Fig. 5 denotes the accuracy of the classifier with varying choices of *N* under the four testing scenes, having the red line indicating the average accuracy. In this figure, we can find that the accuracy reaches the peak when the region size is $N=15$. This is because the road boundary is located in a specific region and large region size for feature extraction would make the classifier confused. Based on this phenomenon, the region sizes after $N=20$ make no sense and this is demonstrated in the figure. In our method, the region size *N* for feature extraction is set to 15.

The second parameter is the feature selection. There are three features in our method, DSIFT, HOG and LBP. All of them reflect the texture cue from a specific aspect and how to fuse them is a remaining question up to now. If fewer features can achieve perfect performance, it is more computationally efficient to use the smaller feature set. For this purpose, different feature sets are evaluated in Fig. 6. It can be seen that DSIFT plays an important role and the features combined with DSIFT can have a good performance. However, the best results are obtained by using the three features altogether. But for faster speed in real applications, only DSIFT suffices for an acceptable performance.

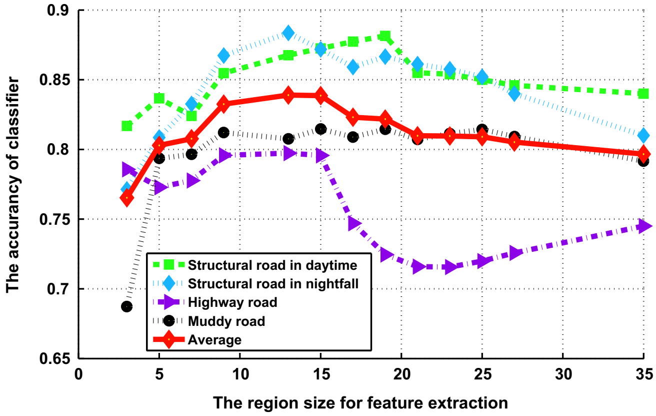


Fig. 5. The performance under different patch size for feature extraction of the sample instance. The horizontal axis is the patch size *N*, and the vertical axis is the accuracy of the classifier. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

The third parameter is the ratio τ between the negative and positive training instances for the model updating procedure. Since τ can serve as the prior probability of different category numbers, its value will affect the trained classifier. In our experiments, the ratio τ is set as follows:

$$\tau \in \left\{1, 1.5, \frac{1}{1.5}, 2, \frac{1}{2}, \ldots, 10, \frac{1}{10}\right\}. \tag{12}$$

The dual pair of $\tau(e.g.,2,1/2)$ makes a balanced choice for possible negative and positive numbers. The evaluation results are shown in Fig. 7. The red line is the average accuracy of the four testing videos. It is obvious that after $\tau=3.5$, the accuracy changes little. Larger τ will not greatly increase the performance. Instead, it may cause the detection rate of boundary (TP) decrease, for the reason that more instances will be determined as negative samples. Therefore, the parameter τ is set to 3.5 in our method.

3.4. Performance analysis

In this part, experimental results are evaluated by qualitative and quantitative means. Typical road detection results of the four sceneries are shown in Fig. 8. From the figure, it is obvious that the proposed method is more accurate and robust in defining the road area and vanishing point. Even for the muddy road with no clear markings, our method can demonstrate a superior performance.

For a more objective comparison, we calculate the statistics of the five measures introduced in Section 3.2. The results are shown in Table 2. We can see clearly that the highest scores are mostly achieved by the proposed method. Only three recall values for the SVM based method are a little better than ours, but the differences are very small. Besides, the other scores of SVM based method is not comparable with ours. Therefore, it is reasonable to claim that the proposed method is more effective than the other competitors.

In the following, a more detailed analysis on the four video sequences will be presented. Since the aforementioned metrics focus on the overall statistics, we will alternatively discuss the results on frame level. For each frame, we first compute its pixel-wise accuracy based on the ground truth. Then we set different thresholds to count the percentage of frames with a higher accuracy above the threshold. Those selected frames are thought to be the correct detections and a larger frame percentage indicates a better performance. By changing the threshold from 0.8 to 1, we can get the statistical curves shown in Fig. 9, from

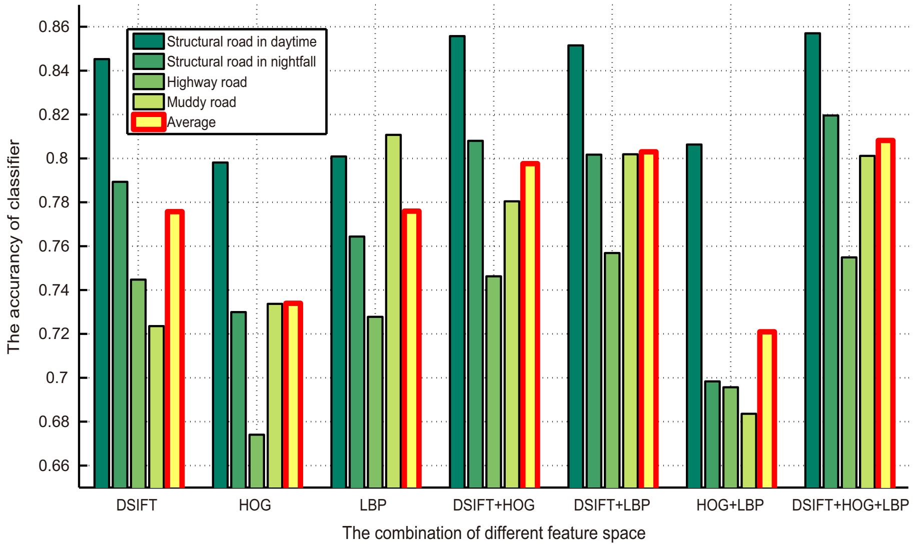


Fig. 6. The performance under different feature groupings. The horizontal axis indicates the various groupings of the features, and the vertical axis represents the accuracy of the classifier.

which the robustness and effectiveness of the three methods can be compared. Intuitively, the higher the curve line and the larger the area below the curve line, the better performance a method has. The same strategy is utilized for vanishing-point evaluation. If the distance between the detected vanishing-point and ground truth is smaller than the threshold, the detection of this frame would be regarded as correct. By varying the threshold from 0 to 20 pixels, the properties of the three methods can also be compared. The obtained results are shown in Fig. 10.

Structural road at daytime: This video is stable and the road boundary has clear color and gradient. Figs. 9(a) and 10(a) show that the proposed method and SVM based method nearly have the same performance. Both of them are far better than the vanishing point based one. In fact, we find the output after SSVM classifier is better than the output by SVM. The subsequent boundary fitting procedure fortunately makes up the gap. Thus the final results demonstrate little difference. But the statistics in Table 2 still tell us that the proposed method is superior to the SVM based one. As for the vanishing point based method, the temporal cue between adjacent frames is not considered, instead of utilizing the previous result as the prior knowledge. Therefore, the obtained results are not stable.

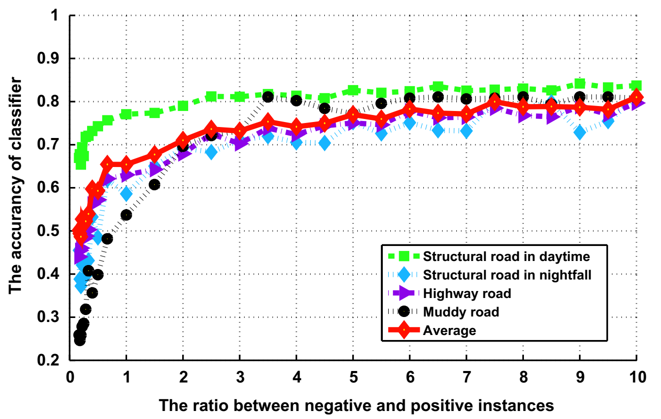


Fig. 7. The performance under different sampling ratio τ. The horizontal axis is the ratio τ according to Eq. (12), and the vertical axis is the accuracy of the classifier. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Structural road at nightfall: The environmental light in this sequence is not adequate and the reflection of neon lights influences the feature extraction. On this occasion, SIFT, HOG and LBP features are all considered so as to obtain a stable inference result of road boundary. The comparisons are shown in Figs. 9(b) and 10(b). It is manifest that the proposed method and SVM-based method are more robust to the cluttered scene than the vanishing point-based method. The reason is mainly due to their targeted sample selection and their use of a reference map. Apart from this, the vanishing-point based method focuses on the gradient information in color space which has a lot noise affected by the light reflection and low contrast, while our method considers the gradient and texture simultaneously, which can adapt to the low contrast environment.

Unstructured highway road: The frames in this sequence have large pitch and yaw and the road boundaries are ambiguous in some frames. Figs. 9(c) and 10(c) show the comparative results. The gradient and texture information of the road boundary change a lot as the vehicle is moving. By utilizing the online-learning strategy that updates the classifier as the new environment appears, the proposed method and SVM-based one are able to tackle it. Meanwhile, since the proposed method considers the underlying structure of the sample instances, it can classify them more precisely. Therefore, it is better than SVM. On the contrary, vanishing point-based method does not consider the structure cue and is unable to adapt to the changes. Thus its performance is poorer.

Unstructured muddy road: This sequence is cluttered by complex texture in the muddy road, which causes the road boundaries

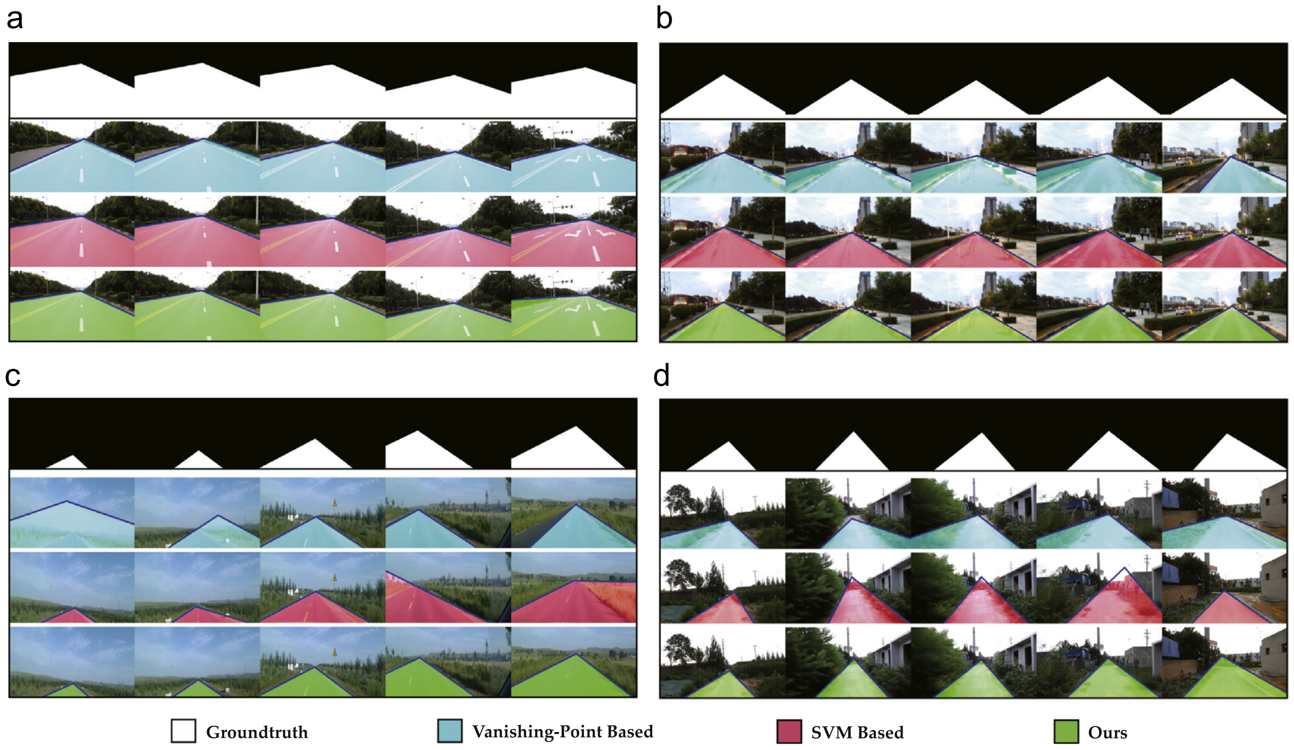


Fig. 8. Typical road detection results of the vanishing-point based, SVM based, and the proposed method in this paper. The four kinds of scenes are separately showed in (a), (b), (c), and (d). For each scene, from top to bottom are the ground truth, the results of two competitors and our method.

to be ambiguous. We define the region between plants and muddy areas as road boundary. The detection results are shown in Figs. 9(d) and 10(d). There are three aspects to explain the good performance of the proposed method. Firstly, although the video frames are cluttered and some unnecessary information far from the boundary may cause the classifier to fail, the targeted instances are sampled along the road boundary which largely improve the judgement of the classifier. Secondly, the online updating strategy makes the classifier adapt to the changing of the road boundary compared to the vanishing-point based method. Thirdly, our method utilizes SSVM to obtain the structure information of the same class which can help a lot for inferring the new instances compared with SVM-based method.

Table 2
Performance of our road detection framework. The data in each column is obtained by computing the average value of the frames in each sequence, and the **Bold** one represents the best.

Methods	Testing video	Accuracy (%)	Precision (%)	Recall (%)	*F*-measure (%)	Distance (*pixel*)
VP [39]	"*srd*" sequence	96.43	95.72	97.34	96.48	9.14
	"*srn*" sequence	87.86	69.69	98.19	81.06	7.40
	"*uhr*" sequence	95.13	80.24	94.30	85.94	5.08
	"*umr*" sequence	89.68	66.68	94.64	77.52	4.82
	average	78.26	78.08	96.12	85.25	6.61
SVM	"*srd*" sequence	99.02	98.20	**99.90**	99.04	2.46
	"*srn*" sequence	96.98	89.88	**99.81**	94.56	3.36
	"*uhr*" sequence	96.30	82.99	**98.26**	89.42	3.57
	"*umr*" sequence	94.27	84.19	88.82	85.37	16.83
	average	96.64	88.82	96.70	92.10	6.56
Ours	"*srd*" sequence	**99.19**	**98.71**	99.69	**99.20**	**2.20**
	"*srn*" sequence	**98.51**	**95.43**	99.16	**97.21**	**3.00**
	"*uhr*" sequence	**99.21**	**98.25**	97.00	**97.57**	**2.06**
	"*umr*" sequence	**99.07**	**96.49**	**98.23**	**97.27**	**3.10**
	average	**99.00**	**97.22**	**98.52**	**97.81**	**2.59**

As mentioned above, online learning strategy helps the model adapt to the variable environments. Moreover, structure information is utilized to improve the performance of the classifier and mitigate the fact that an online learning strategy can degrade the model a lot if previous detection results are incorrect. Thus the structure information and the online learning complement each other, each correcting the other's errors. This conclusion can be generated from the detection results of the four testing videos, especially for the last two videos. To be specific, the camera for capturing highway road scenes have significant jitters aiming at simulating complex road scene. The muddy road scenes are chosen to verify the situation where the road boundary is ambiguous. But under these two situations, our method can still handle the unstructured road detection very well.

3.5. Discussion

In this section, we present two aspects for further discussion. The first one is about the assumption on the road boundary type. The second one is about the computational time. The details are discussed in the following.

(1) *Adaptability for road boundary type*: In this paper, we have made an assumption that the road boundary can be fitted by a straight line. The starting point of making such assumption can be explained in two aspects. Firstly, straight road boundary often occurred and some non-straight road boundary can be regarded as straight when it has small curvature. This phenomenon is even more obvious under the circumstance that the road region almost fully fills the field of view. Secondly, the analysis of curved boundary follows a similar way as what has been discussed in

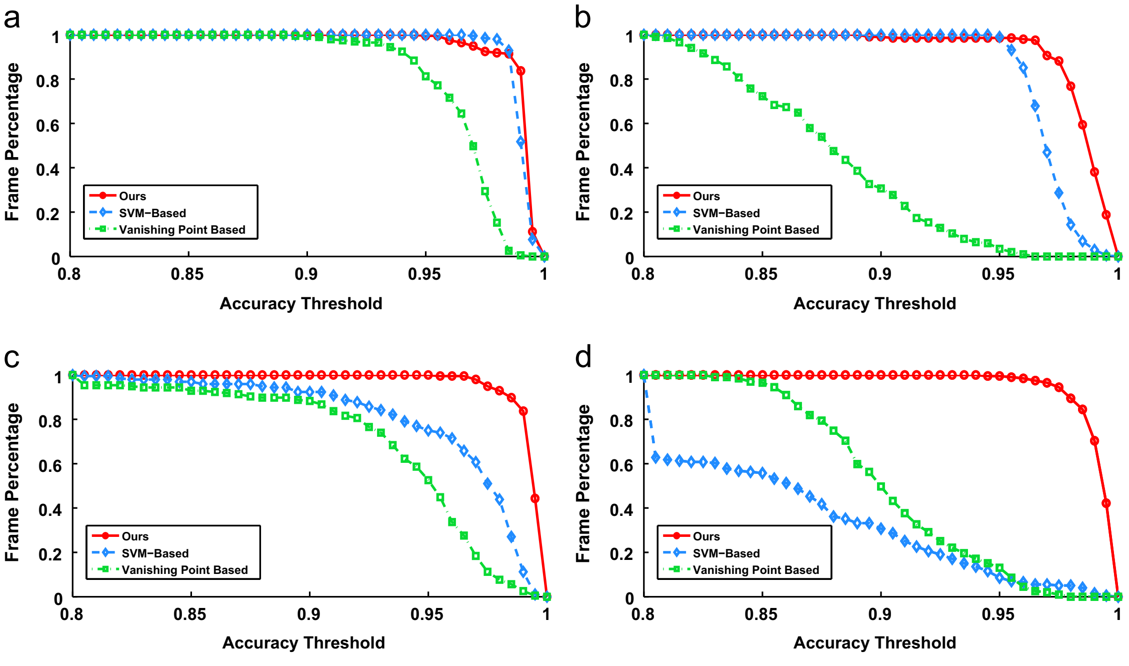


Fig. 9. Evaluation of the proposed method with two competitors by *road region accuracy*. After road detection, each frame has an accuracy according to the ground truth labelings. Then we set different thresholds (horizontal axis) to count the percentage of frames (vertical axis) with a higher accuracy above the threshold. (a)–(d) are respectively the results of the four road scenes.

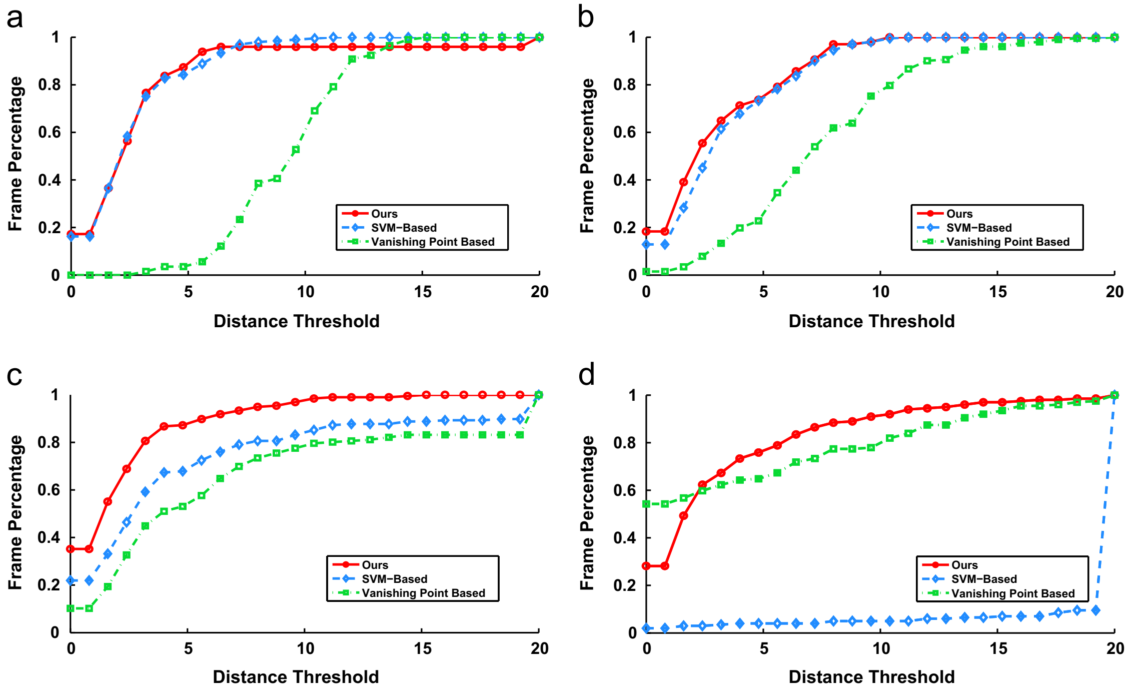


Fig. 10. Evaluation of the proposed method with two competitors by *vanishing point distance*. After road detection, each detected vanishing point has a distance from the ground truth one. Then we set different thresholds (horizontal axis) to count the percentage of frames (vertical axis) with a smaller distance than the threshold. (a)–(d) are respectively the results of the four road scenes.

the straight boundary occasion. The only difference is the boundary fitting procedure, in which case we only need to employ a curve fitting function. This is not a difficult task. From this aspect, our assumption is reasonable and the proposed framework can be readily extended to the curved boundary.

(2) *Computational complexity*: In our experiments, the proposed method is implemented in MATLAB on the platform of Microsoft Windows with Inter i3 3.4 GHz, 2 GB memory without any specific code optimization. The average running time of each frame is 14 s and nearly 36% of the timing consumption is in the feature extraction procedure. However, the main purpose of road detection is for real-time driver assistance or robot navigation and efficiency is a critical factor. One solution is to use fewer features while maintaining an acceptable performance. For example, if only SIFT feature is employed, the average time will be reduced to about 8 s. The second solution is by hardware or software speedup. Parallel computation (e.g., feature extraction and SSVM retraining can be conducted at the same time) and GPU are possible choices [42].

4. Conclusion

In this paper, we present an online-learning method for efficiently exacting the drivable road region in a video sequence. Firstly, the targeted sampling instances are selected within the boundary neighborhood. Then a fusion of features are used to describe the extracted instances. After that, the feature vectors are input to a structure SVM classifier to determine their binary label of boundary and non-boundary, followed by a fitting procedure to enhance the reliability of the detected boundary. Finally, the road area is inferred from the boundary and the learned classifier is updated online to ensure its adaptability to the changing environment. The superiority of the proposed method is verified on the data set collected by ourselves and the experimental results show that it outperforms the other competitors.

In the future, we plan to tackle more complex situations in real life. For example, when the road shape is changeable or the illumination is varying, how to estimate the road region is still a challenging problem. Besides, the GPU implementation of the proposed method is also desired to make the system practical for real-time implementation.

Acknowledgments

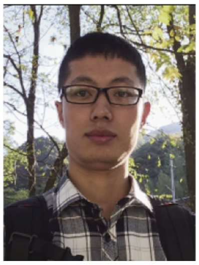
This work is supported by the National Basic Research Program of China (Youth 973 Program) (Grant no. 2013CB336500), State Key Program of National Natural Science of China (Grant no. 61232010), National Natural Science Foundation of China (Grant nos. 61172143, 61379094, and 61105012), Natural Science Foundation Research Project of Shaanxi Province (Grant no. 2015JM6264), Fundamental Research Funds for the Central Universities (Grant no. 3102014JC02020G07), and Open Research Fund of Key Laboratory of Spectral Imaging Technology, Chinese Academy of Sciences.

References

[1] Global Status Report on Road Safety, World Health Organization, Geneva, Switzerland, 2013. ⟨http://www.who.int/violence_injury_prevention/road_safety_status/2013/en/index.html.⟩.
[2] X. Mo, V. Monga, R. Bala, Z. Fan, Adaptive sparse representations for video anomaly detection, IEEE Trans. Circuits Syst. Video Technol. 99 (2013) 1.
[3] P. Borges, N. Conci, A. Cavallaro, Video-based human behavior understanding: a survey, IEEE Trans. Circuits Syst. Video Technol. 23 (11) (2013) 1993–2008.
[4] K. Muller, A. Smolic, M. Drose, P. Voigt, T. Wiegand, 3-D reconstruction of a dynamic environment with a fully calibrated background for traffic scenes, IEEE Trans. Circuits Syst. Video Technol. 15 (4) (2005) 538–549.
[5] J. McCall, M. Trivedi, Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation, IEEE Trans. Intell. Transp. Syst. 7 (1) (2006) 20–37.
[6] H. Cheng, B. Jeng, P. Tseng, K. Fan, Lane detection with moving vehicles in the traffic scenes, IEEE Trans. Intell. Transp. Syst. 7 (4) (2006) 571–582.
[7] S. Wu, H. Chiang, J. Perng, C. Chen, B. Wu, T. Lee, The heterogeneous systems integration design and implementation for lane keeping on a vehicle, IEEE Trans. Intell. Transp. Syst. 9 (2) (2008) 246–263.
[8] A. Huang, D. Moore, M. Antone, E. Olson, S. Teller, Finding multiple lanes in urban road networks with vision and lidar, Auton. Robots 26 (2–3) (2009) 103–122.

[9] X. Cao, R. Lin, P. Yan, X. Li, Visual attention accelerated vehicle detection in low-altitude airborne video of urban environment, IEEE Trans. Circuits Syst. Video Technol. 22 (3) (2012) 366–378.
[10] A. Broggi, P. Cerri, S. Ghidoni, P. Grisleri, H. Jung, A new approach to urban pedestrian detection for automatic braking, IEEE Trans. Intell. Transp. Syst. 10 (4) (2009) 594–605.
[11] C. Pai, H. Tyan, Y. Liang, H. Liao, S. Chen, Pedestrian detection and tracking at crossroads, Pattern Recognit. 37 (5) (2004) 1025–1034.
[12] L. Oliveira, U. Nunes, P. Peixoto, M. Silva, F. Moita, Semantic fusion of laser and vision in pedestrian detection, Pattern Recognit. 43 (10) (2010) 3648–3659.
[13] A. Hillel, R. Lerner, D. Levi, G. Raz, Recent progress in road and lane detection: a survey, Mach. Vis. Appl. (2012) 1–19.
[14] C. Guo, S. Mita, D. McAllester, Robust road detection and tracking in challenging scenarios based on Markov random fields with unsupervised learning, IEEE Trans. Intell. Transp. Syst. 13 (3) (2012) 1338–1354.
[15] F. Oniga, A. Nedevschi, M. Meinecke, T. Thanh-Binh, Road surface and obstacle detection based on elevation maps from dense stereo, in: Proceedings of IEEE Conference on Intelligent Transportation Systems, 2007, pp. 859–865.
[16] A. Sappa, F. Dornaika, D. Ponsa, D. Gerónimo, A. López, An efficient approach to onboard stereo vision system pose estimation, IEEE Trans. Intell. Transp. Syst. 9 (3) (2008) 476–490.
[17] B. Fardi, G. Wanielik, Hough transformation based approach for road border detection in infrared images, in: Proceedings of IEEE Intelligent Vehicles Symposium, 2004, pp. 549–554.
[18] A. Borkar, M. Hayes, M. Smith, Robust lane detection and tracking with RANSAC and kalman filter, in: Proceedings of IEEE Conference on Image Processing, 2009, pp. 3261–3264.
[19] H. Sawano, M. Okada, Road extraction by snake with inertia and differential features, in: Proceedings of International Conference on Pattern Recognition, vol. 4, 2004, pp. 380–383.
[20] R. Maurya, P. Gupta, A. Shukla, Road extraction using *k*-means clustering and morphological operations, in: Proceedings of International Conference on Image Information Processing, 2011, pp. 1–6.
[21] C. Rotaru, T. Graf, J. Zhang, Color image segmentation in HSI space for automotive applications, J. Real-Time Image Process. 3 (4) (2008) 311–322.
[22] F. Oniga, S. Nedevschi, M. Meinecke, T. To, Road surface and obstacle detection based on elevation maps from dense stereo, in: Proceedings of IEEE Intelligent Transportation Systems Conference, 2007, pp. 859–865.
[23] Y. He, H. Wang, B. Zhang, Color-based road detection in urban traffic scenes, IEEE Trans. Intell. Transp. Syst. 5 (4) (2004) 309–318.
[24] M. Sotelo, F. Rodriguez, L. Magdalena, Virtuous: vision-based road transportation for unmanned operation on urban-like scenarios, IEEE Trans. Intell. Transp. Syst. 5 (2) (2004) 69–83.
[25] J. Alvarez, A. Lopez, Road detection based on illuminant invariance, IEEE Trans. Intell. Transp. Syst. 12 (1) (2011) 184–193.
[26] M. Foedisch, A. Takeuchi, Adaptive real-time road detection using neural networks, in: Proceedings of IEEE Conference on Intelligent Transportation Systems, 2004, pp. 167–172.
[27] T. Son, S. Mita, A. Takeuchi, Road detection using segmentation by weighted aggregation based on visual information and a posteriori probability of road regions, in: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 2008, pp. 3018–3025.
[28] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[29] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2005, pp. 886–893.
[30] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on featured distributions, Pattern Recognit. 29 (1) (1996) 51–59.
[31] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.
[32] H. Xue, S. Chen, Q. Yang, Structural support vector machine, in: Advances in Neural Networks, 2008, pp. 501–511.
[33] T. Joachims, T. Finley, C. Yu, Cutting-plane training of structural SVMS, Mach. Learn. 77 (1) (2009) 27–59.
[34] X. Ren, J. Malik, Learning a classification model for segmentation, in: Proceedings of IEEE Conference on Computer Vision, 2003, pp. 10–17.
[35] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, SLIC superpixels compared to state-of-the-art superpixel methods, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2274–2282.
[36] M. Fischler, R. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.
[37] P. Torr, A. Zisserman, MLESAC: a new robust estimator with application to estimating image geometry, Comput. Vis. Image Underst. 78 (1) (2000) 138–156.
[38] Y. Freund, R. Schapire, A desicion-theoretic generalization of on-line learning and an application to boosting, in: Proceedings of Computational Learning Theory, 1995, pp. 23–37.
[39] H. Kong, J. Audibert, J. Ponce, Vanishing point detection for road detection, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 96–103.
[40] Q. Wang, Y. Yuan, P. Yan, X. Li, Saliency detection by multiple-instance learning, IEEE Trans. Cybern. 43 (2013) 660–672.
[41] Q. Wang, Y. Yuan, P. Yan, Visual saliency by selective contrast, IEEE Trans. Circuits Syst. Video Technol. 23 (7) (2013) 1150–1155.
[42] S. Akramullah, I. Ahmad, M. Liou, Performance of software-based MPEG-2 video encoder on parallel and distributed systems, IEEE Trans. Circuits Syst. Video Technol. 7 (4) (1997) 687–695.

Yuan is currently a Full Professor with the Chinese Academy of Sciences, Beijing, China. She has authored or coauthored over 150 papers, including about 100 in reputable journals such as IEEE Transactions and Pattern Recognition, as well as conference papers in CVPR, BMVC, ICIP, and ICASSP. Her current research interests include visual information processing and image video content analysis.



Zhiyu Jiang is currently working toward the Ph.D. degree in the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China. His current research interests include artificial intelligence and computer vision.

Qi Wang received the B.E. degree in automation and Ph.D. degree in pattern recognition and intelligent system from the University of Science and Technology of China, Hefei, China, in 2005 and 2010 respectively. He is currently an associate professor with the School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL) , Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.