Computers and Electrical Engineering 000 (2017) 1–9

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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

Combining CNN and MRF for road detection☆

Lei Geng[a,b], Jiangdong Sun[a,b], Zhitao Xiao[a,b,*], Fang Zhang[a,b], Jun Wu[a,b]

[a] School of Electronics and Information Engineering, Tianjin Polytechnic University, Tianjin, China
[b] Tianjin Key Laboratory of Optoelectronic Detection Technology and Systems, Tianjin, China

A R T I C L E I N F O

Article history:
Received 14 July 2017
Revised 20 November 2017
Accepted 20 November 2017
Available online xxx

Keywords:
Driver assistance system
Road detection
Super-pixel
CNN
MRF

A B S T R A C T

Road detection aims at detecting the road surface ahead of the vehicle and plays a crucial role in driver assistance systems. To improve the accuracy and robustness of road detection approaches in complex environments, a new road detection method based on a convolutional neural network (CNN) and Markov random field (MRF) is proposed. The original road image is segmented into super-pixels of uniform size using the simple linear iterative clustering (SLIC) algorithm. On this basis, we train the convolutional neural network, which can automatically learn the features that are most beneficial to the classification. The trained convolutional neural network (CNN) is then applied to classify road and non-road regions. Finally, based on the relationship between the super-pixel neighborhood, we utilize Markov random field (MRF) to optimize the classification results of the convolutional neural network (CNN). The approach provides the better performance.

1. Introduction

Research shows that about 1.4 million people die in road traffic accidents globally every year, owing to driver fatigue and inattention [1]. Science fiction portrays artificial intelligence (AI) as robots with human-like characteristics. However, what we can achieve today is weak AI, which is designed to perform special tasks [2], e.g., motor anomaly detection for unmanned aerial vehicles [3] and advanced driver assistant systems (ADASs) [4]. An ADAS can reduce the incidence of traffic accidents by notifying and guiding the driver. Road detection based on vision is the key to a driver assistance system, which can provide clues for obstacle detection and help with path planning [4].

Roads can generally be divided into structured and unstructured roads. Structured roads have clear road marking lines, clear road borders, and special color information, and include highways and city roads. The problem of structured road detection can be simplified as the problem of road marking line detection. On the contrary, unstructured roads do not have obvious road marking lines or road borders, and thus detection methods are still in the research stage. Road and non-road regions are more difficult to distinguish because a road environment is complex and changeable, and is influenced by light, water spots, shadows, complex obstacles, and other factors.

At present, road detection methods can be divided into two categories. The first method is based on modeling. For instance, Wang et al. [5] convert an image into an inverse perspective space and preprocess the image using a hybrid Gaussian anisotropic filter. They then use a Bezier spline curve to construct a variable road template. Finally, they detect the

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Associate Editor Dr. Huimin Lu.

* Corresponding author at: School of Electronics and Information Engineering, Tianjin Polytechnic University, Tianjin, China.
E-mail addresses: genglei@tjpu.edu.cn (L. Geng), sunjiangdong@stu.tjpu.edu.cn (J. Sun), xiaozhitao@tjpu.edu.cn (Z. Xiao), zhangfang@tjpu.edu.cn (F. Zhang), wujun@tjpu.edu.cn (J. Wu).

https://doi.org/10.1016/j.compeleceng.2017.11.026

road using the improved RANSAC algorithm to solve the template parameter. Wang et al. [6] combine a parabolic model with a Hough transform to detect a road boundary. They first utilize the Hough transform to fit the edge points, and then use the parabolic model to fit the curved road boundary. Kong et al. [7] utilize a multi-scale Gabor wavelet transform to calculate the texture direction of the pixels, and estimate the vanishing point of the road through adaptive voting. Finally, they utilize a vanishing point to constrain the edge detection, and then obtain the road boundary. Such methods can be used to detect structured roads accurately and achieve relatively complete detection results. However, based on actual movements of the vehicles, the shape of the road constantly changes. Establishing a robust model for road matching is a challenge problem. The most common methods are from the second category of road detection, which is based on features. In this aspect, scholars at home and abroad have conducted a number of studies. For instance, Alvarez et al. [8] focused on road detection under different lighting conditions. They use camera parameters to restore the illuminant-invariant feature space, and then apply the road modeling according to the gray-scale road value. The pixels are labeled by combining a model-based classifier. Mendes et al. [9] proposed a network for road detection that takes advantage of a large contextual window and uses a Network-in-Network (NiN), and a multi-layer convolutional neural network (CNN) architecture with a new loss function for detecting free space. Brostow et al. [10] integrate the appearance features and structure cues obtained from 3D point clouds, and then use a randomized decision forest to find the road regions. A framework proposed by Sturgess et al. [11] combines motion and appearance features. A conditional random field (CRF) is then employed to integrate higher-order potentials and other features. By minimizing the energy function, the labels are obtained. Yuan et al. [12] extracted HOG, LBP, and DSIFT features of the area around a road border to train an SSVM classifier, and then obtain a series of sparse points, which are the candidates of the road boundaries. Finally, they fit the boundary using the RANSAC algorithm. Fernández et al. [13] make use of a watershed transformation to segment an original image into super-pixels, and then extract the color and texture features of the super-pixels to train the decision-tree based classifiers to complete the road detection. Alvarez et al. [14] combine a CNN with a color-space based texture descriptor, and label the pixels by comparing the multiplied probability with the threshold. Lu et al. proposed a fast level set model based method for intensity inhomogeneity correction [15] and a spectral-property based color correction method [16], which can detect the contours at weak or blurred edges efficiently. Such methods mainly use the color, texture, and edge features to detect a road, and are not sensitive to the shape of the road and are thus suitable for any road shape. However, they are easily influenced by shadows or light when a road scene is complex and changeable, making it more difficult to choose the features.

In summary, to improve the robustness of a road detection method in a complex environment, we propose a new approach integrating a CNN with a Markov random field (MRF) for road detection. The main contributions of this study are as follows: (i) the design of a novel network structure that is suitable for road detection, and (ii) label optimization using an MRF model to improve the accuracy of the detection. The overall accuracy of our method based on the Cambridge-driving Labeled Video Database (CamVid) [17] can reach up to 96.1%. The remainder of this paper is organized as follows: In Section 2, we discuss the main aspect of the proposed method. Section 2.1 introduces why we use super-pixels and how many super-pixels we should choose. The details of the CNN are introduced in Section 2.2, and the MRF algorithm used to optimize a label is given in Section 2.3. Section 3 describes the details of the database and the network structure. Different experiments conducted using the database are also described. Finally, some concluding remarks and directions for future work are provided in Section 4.

2. Road detection

2.1. Super-pixel extraction

Super-pixels can be used to extract the local features, obtain more efficient structural information, and reduce the computational complexity of the subsequent processing better than a normal pixel, which is the basic unit used in traditional processing methods. A single super-pixel usually describes only one target, avoiding the confounding targets found in a rectangular-based window. Compared with other super-pixel segmentation algorithms such as Normalized-Cuts, Graph-Cuts, Turbo-Pixel, Quick-Shift, and Mean-Shift [18], the simple linear iterative clustering (SLIC) algorithm has the advantages of a regular region shape through pre-segmentation, less time consumption, and a better preservation of the target boundary. In this paper, the SLIC algorithm is used to obtain the super-pixels.

The SLIC algorithm is a super-pixel segmentation method based on K-means clustering that uses the color similarity and positional relationship of the pixels to generate super-pixels [19]. For each super-pixel, the three-dimensional color features l, a, and b in the CIELAB color space, and the two-dimensional position information x and y, are employed to describe the center. The concrete steps are as follows:

Initialize the cluster centers. Assume that super-pixel K with a desired number of approximately equally sized pixels is used as input. Thus, for an image with N pixels, the approximate size of each super pixel is N/K pixels. A super-pixel center will occur at every grid interval $s = \sqrt{N/K}$. The cluster centers are moved to the lowest gradient position in a 3×3 neighborhood to avoid placing them at an edge and thereby reducing the chance of choosing a noisy pixel. The corresponding labels are then assigned for all cluster centers.

Similarity measure. Search for the best matching pixels from a $2S \times 2S$ square neighborhood around the cluster center. Assign the label of the cluster center to the best-matched pixels. We then iteratively repeat the process until convergence