Pattern Recognition Letters 108 (2018) 1–7

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Ground segmentation and free space estimation in off-road terrain

Mahmoud Hamandi, Daniel Asmar*, Elie Shammas

Vision and Robotics Lab, American University of Beirut, Bliss Street, Beirut, Lebanon

ARTICLE INFO

Article history:
Received 22 June 2017
Available online 26 February 2018

MSC:
41A05
41A10
65D05
65D17

ABSTRACT

In this paper, we propose a novel approach for ground segmentation and free space estimation of outdoor environments. The system is completely self-supervised and relies on two modules: the first module is built around a Fully Convolutional Network (FCN), and is used for ground segmentation after the system is initiated. The second module relies on depth information paired with interactive graphs cuts, and is used to train the FCN at startup, and anytime the FCN's performance degrades during runtime. This usually happens when the camera observes a new type of outdoor scene, which is foreign to the FCN. Experiments were conducted on three datasets of different ruggedness to highlight the advantages of the proposed method.

1. Introduction

With the era of autonomous vehicles about us, scene understanding is becoming ever more important. For autonomous navigation, one of the most fundamental scene understanding tasks is to delineate free space from obstacles, thereby allowing the vehicle to drive autonomously, and plan its path along its journey.

Previous work on free space segmentation employ a variety of sensors such as active LIDARs and RADARs [7,20]. Because of their relatively low cost, and the high bandwidth of information they offer, vision-based solutions are also very popular for free space estimation. Marks et al. [16] trained an offline classifier to segment ground in a short range stereo camera. Then they used the segmented ground as training data to segment longer range ground in a monocular image. Their method requires human supervision, and is not guaranteed to work in unseen environments that differ from the training data.

Kühnl et al. [13] proposed to combine confidence maps generated by three classifiers that detect road, boundary, and lanes, and accordingly segment monocular images of a road. However, their approach requires all three road elements to be present in each query image.

Alvarez and Lopez [2] compute illuminant invariant images from the captured monocular ones, then assume that pixels at the root of the image can be used as training data to label the rest of the pixels. While this works for monotone roads, free space with more variant representations require a more elaborate training data extraction algorithm.

Recently, Brust et al. [6] proposed a new technique to train a Convolutional Neural Network (CNN) for pixel-wise ground segmentation. The proposed technique processes each patch of the image with a CNN and then classifies the central pixel of that patch. In addition, Brust *et al.*noticed that the pixel's position is a strong prior for the labeling, and as such incorporated the spatial prior as an input to the fully connected layers of the CNN. Although they showed promising results, their algorithm requires supervised human interaction in the form of hand labeled training data. As a solution, Sanberg et al. [19] extracted a self-supervised weak classifier to train the network, and proved that online training of the network with the weak classifier can outperform both the weak classification and the offline trained network. However, their method requires high runtime because it requires online training of the network for each input image after extracting its weak classifier.

Similarly, Alvarez et al. [1] train a semantic segmentation CNN with machine-generated labels on a generic dataset. Then combine the segmentation of the offline trained CNN with an online extracted texture based classifier, using a Positive Naïve Bayes framework.

The stated methods exploit either intensity data, or almost only depth data; however, stereo sensors allow one to exploit the complementarity of pixel intensity with object depth to produce robust segmentation solutions.

Badino et al. [3] proposed the Stixel World representation, where the stereo image is first transformed into its polar coordinate representation. Then after calculating the occupancy of each pixel, final ground and obstacles are estimated using dynamic programming that imposes spatial smoothness between adjacent pixels. Finally, obstacles are grouped into stixels, to provide a higher

* Corresponding author.
E-mail address: da20@aub.edu.lb (D. Asmar).

https://doi.org/10.1016/j.patrec.2018.02.019
0167-8655/© 2018 Published by Elsevier B.V.

order representation of the space, thus allowing simpler further processing of the obstacles. Many variations of the stixel world algorithm exist, such as the ones presented in [4,18].

Vernaza et al. [21] used a stereo sensor in a Markov Random Field framework to classify pixels in the image belonging to the ground plane. The largest planar region is assumed to be the ground plane, and pixels belonging to it are taken as ground pixels. Hadsell et al. [9] use the Hough transform up to three times to fit planes to stereo point clouds, while bounding the maximum slope a UGV can drive on, to find the points belonging to the ground plane. Konolige et al. [12] use Ransac plane fitting to determine points belonging to the largest plane in 3-D stereo generated point clouds, which is assumed to be the ground plane. These points are then used to supervise the classification of far away pixels in a monocular image.

These training data extraction methods fail in scenarios where the ground plane is not the largest plane in the image. In contrast, we propose in this paper a more robust method for extracting training data for our machine learning solution. The method relies on first processing stereo images to produce the *v*-disparity and *u*-disparity images [14], which serve in the delineation of the ground plane.

Recently, deep learning solutions have become popular approaches for semantic segmentation, where fully connected layers, and deconvolutional networks have been designed to segment scenes of arbitrary sizes [15,17]. However, these algorithms, in addition to their protracted training time, might fail when confronted with unseen objects.

In this paper, we propose a multi-level ground segmentation system: at a first level, training data for ground pixel are identified using a self-supervised technique. At a second level, a deep network uses the data from the first level for training. The deep network is designed to adapt to different environments by relying on a weak classifier, which re-trains when its performance drops.

The contributions of this paper are as follows:

- The development of a novel technique to extract reliable ground patches in a stereo image pair without any offline training. The method is self-supervised, near real-time, and operates in structured and unstructured environments.
- An adaptive learning technique based on the decay of network performance.
- Evaluation of the proposed technique, and comparison against the most relevant self-supervised techniques in the literature.

The remainder of the paper is structured as follows: Section 2 provides the details of the proposed data extraction method, followed by the adaptive technique for online ground segmentation. Section 3 describes the experiments, and benchmarks our system to the state of the art. Finally, Section 4 concludes the paper.

2. Architecture for ground segmentation

In this section, we explain the components of the system architecture. A Fully Convolutional Network (FCN) lies at the core of the segmentation (Fig. 1). The network is first trained offline at system start-up, to match the labels generated by first-stage classifier (referred to as FSGC and discussed in Section 2.5). The network's labels are then used online as ground data while being assessed with a weak classifier (referred to as PNB-RGBD). Both the training and the weak classifiers are detailed in Section 2.5.4. The network is re-trained online whenever the FCN and the weak classifications do not match.

2.1. *Fully convolutional network*

A Fully Convolutional Network (FCN) [15] is a Convolutional Neural Network (CNN) in which all fully connected layers are replaced with convolutional layers, thereby turning the network into a deep filter, without any decision layers. After being trained, the FCN can be used for segmentation by classifying each pixel into a category [15]. We opt for the FCN introduced by Brust et al. [6], implemented in the CN24 framework. The network architecture is as follows: Conv ($7 \times 7 \times 12$); MaxP (2×2); ReLU; Conv ($5 \times 5 \times 6$); ReLU;Full(48x); ReLU;Full(192x); spatial prior; ReLU;Full(1x)+tanh. The CN24 library replaces the fully connected layers with their convolutional counterparts. The spatial prior was added by Brust et al. [6], where the normalized row position of each pixel is added as an additional queue to the fully connected layers. The row position of the pixel can provide a strong prior of the pixel label, and they proved its advantage in ground segmentation. The above CNN architecture treats each pixel, with its neighbor, as a training point; thus for one image, we have an abundance of training data. While this is usually a good thing, the large amount of data from a single image can lead to over-fitting.

2.2. *Self-supervised training*

Self-supervised training is the process by which a first classifier (usually weak), supervises the training of a second classifier, which usually outperforms the weak classifier [19], while exploiting data queues complementary to the weak one.

Although it is computationally less expensive to train one classifier offline and employ it to segment all ground representations, the variety of spaces the vehicle might traverse would drive the classifier to find a compromise between all representations, and thus reduces its efficacy for most. The self-supervised training approach adopted here can provide a classifier that is specific for each scene, without the need to hand label thousands of images.

We introduce a technique for Free Space estimation using Graph Cuts (hereafter named FSGC and detailed in Section 2.5), which is capable of training the FCN when required. To reduce the computational requirements of the system, the training is performed infrequently. FSGC has an advantage over other algorithms from the literature, as it can classify every pixel based on its intensity information while combining depth and color queues; other algorithms classify ambiguous pixels as obstacles or provides the nearest label.

2.3. *Offline training*

During system launch, the software acquires a small number of images, which are segmented using FSGC, and the resulting labels are used to train the FCN. As long as the ground representation does not change, the resulting FCN is employed for ground segmentation; however, if it does, the FCN is fine-tuned online as shown in Section 2.4.

2.4. *Online segmentation and training*

After the training of the FCN offline, the network is employed online to segment ground in the newly acquired images as shown in Fig. 1. In addition, each image is segmented using a weak classifier, assisted by its stereo data. Then the FCN segmentation is assessed for precision and recall, by assuming the weak classifier as ground truth.

The precision and recall calculated would reflect the compatibility between the two classifications. The lack of compatibility (*i.e.,* low precision and recall), suggest that at least one of the two classifiers cannot label the present scene correctly, thus the FCN