2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Robust Road Detection from a Single Image

Junkang Zhang*, Siyu Xia*, Kaiyue Lu†, Hong Pan* and A. K. Qin‡
*Key Laboratory of Measurement and Control of Complex Systems of Engineering, Southeast University, Nanjing, China
†College of Engineering and Computer Science, Australian National University, Canberra ACT 2601, Australia
‡School of Computer Science and Information Technology, RMIT University, Melbourne VIC 3001, Australia
jkzhang@seu.edu.cn, xsy@seu.edu.cn, luke213112354@gmail.com, mspanhong@hotmail.com, qfred008@gmail.com

***Abstract*—Road detection from images is a challenging task in computer vision. Previous methods are not robust, because their features and classifiers cannot adapt to different circumstances. To overcome this problem, we propose to apply unsupervised feature learning for road detection. Specifically, we develop an improved encoding function and add a feature selection process to obtain robust and discriminative road features. Besides, a road segmentation algorithm is proposed to extract road regions from the learned feature maps, in which a tree structure is established to represent the hierarchical relations of various regions segmented by multiple thresholds, and a two-loop optimization is then employed to select the most stable regions as road areas. Experimental results on several challenging datasets justify the effectiveness of our method.**

I. Introduction

Vision-based road detection is a critical yet challenging task for ADAS (Advanced Driver Assistance Systems). Since visual data can provide rich information about driving scenarios, vision-based systems have great potential in comprehensive road scene understanding. However, current vision-based systems are far from being developed, many problems such as instability and inefficiency in feature representation of roads all cause obstacles for their practical application in real driving. In this paper, we will focus on road area detection from a single image captured by a front monocular camera.

As aforementioned, lack of robustness is the major shortcoming of existing road detection algorithms. One reason for this is lack of adaptable road representations. In previous works, multiple cues including colors [1], [2], vanishing point [3], [4], shape [5], and their combinations [6] have been explored, while most of these hand-designed features are sensitive to the variety in circumstances like shadows, underexposure, and occlusions, etc. Besides, there are road features learned from deep convolutional networks [7], [8] which fit to training data but might not adapt to unseen scenarios. So, it is expected to automatically learn universal representations for road detection.

Recently, much attention has been paid to unsupervised feature learning, and K-means clustering algorithm has been justified as a fast and effective method in learning new representations [9]–[11]. Particularly, in [11], K-means learned features directly from image patches and demonstrated superior performance in object recognition tasks over other unsupervised feature learning methods. So, it is natural to apply K-means in our task to learn discriminative and robust road features.

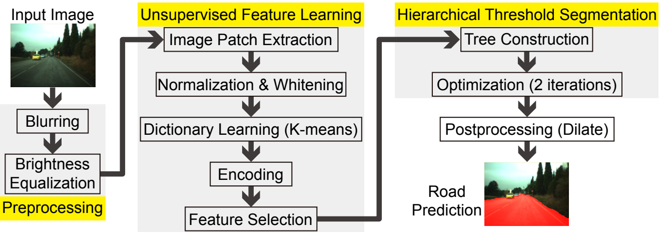


Fig. 1: Algorithmic framework of our method.

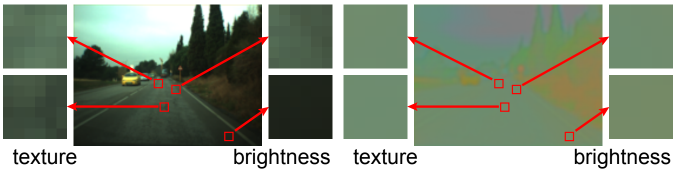
In our previous work [12], we exploited K-means for road feature learning, in which the framework of [11] was directly plugged into our method. In contrast, in this paper, we have a deeper look at this framework, and improve the discriminability of the learned road features by modifying the feature encoder and adding a feature selection process. In addition, we propose a hierarchical threshold segmentation algorithm for road area extraction. In particular, we first use a tree structure to represent the hierarchical relations of segmented regions by multiple thresholds, and then find the most ”stable” tree nodes as the road area through a two-loop optimization process. Our method is based on a simple assumption that, the closer to the horizontal middle and the bottom of the image, the higher probability a pixel will have of belonging to roads.

II. Road Detection Algorithm

Our road detection algorithm consists of three stages which are illustrated in Fig. 1. First, the input road image is preprocessed via blurring and brightness equalization. Then, in the unsupervised feature learning stage, a feature dictionary is learned from processed patches extracted from the image, and encoded features are selected for the next stage. Finally, in the segmentation stage, a threshold tree is constructed over the feature maps. and the most stable tree nodes are selected as road regions via a two-loop optimization process.

A. Image Preprocessing

First of all, blurring and brightness equalization are employed to preprocess the road image for learning better features. On the one hand, there always exists texture diversity in road surfaces caused by their varying distances to the camera, which makes it difficult to obtain representative features for the entire road area. To deal with this problem, image blurring is adopted to remove such road texture variations. On the other hand, there often exists brightness variation in road surfaces



(a) Without preprocessing. (b) After preprocessing.

Fig. 2: A comparison on brightness and texture among different road areas (a) without and (b) after preprocessing.

due to reflection of sunlight from different angles, which is also unfavorable to learning representative road features. To fix this problem, brightness equalization is also applied to balance the luminance of the road surface. Fig. 2 compares road patches in different areas without and after preprocessing.

For blurring, we convolve the image with a low-pass Gaussian filter

$$G(i,j,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}} \tag{1}$$

where σ is the standard deviation of Gaussian distribution, and i and j are horizontal and vertical distances (in pixels) from a pixel to the filter center respectively.

For brightness equalization, we use Eq. (2) to reduce brightness variations

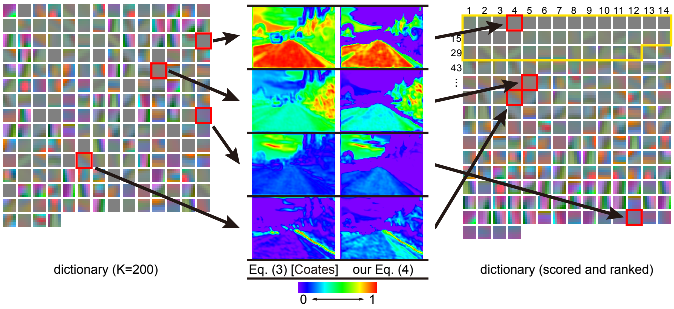
$$I_{eq}(x,y,c) = \frac{I(x,y,c)}{\sum\limits_{i}\sum\limits_{j} I_{gray}(x+i,y+j)G(i,j,\sigma)} \tag{2}$$

where I is an input color image; I_{gray} is the grayscale image converted from I; I_{eq} is the equalized image; and x, y and c are index values of pixels in a color image with 3 channels. Actually, the denominator of Eq. (2) is I's brightness map which is estimated by blurring I_{gray} using a Gaussian filter whose σ is set to 0.5 empirically.

B. Unsupervised Feature Learning and Feature Selection

Remember that color-based road features extracted directly from an image are not robust enough to various road situations. So, it is desirable to convert the road image into robust and discriminative representations to distinguish roads from non-road areas. Inspired by the works in [11], we employ K-means for unsupervised feature learning from pixels in a single road image. However, we found that the algorithmic pipeline in [11] has limitations in our road detection task which will be detailed below. In addition, to tackle with the limits of the pipeline in [11] for our road detection task, we further propose two major improvements over the original framework, i.e., introducing a new encoding method, and adding a feature selection operation.

In the beginning, our method extracts n image patches with size $w \times w \times ch$ from every possible locations of the input image I_{eq}, where w defines patch size and ch is the number of color channels. For a 160×120 image, if $w = 6$, then $n = (160-6+1) \times (120-6+1) = 17825$. Each patch is represented by a d-dimensional vector $\mathbf{x}^{(i)} \in \mathbb{R}^d, i = 1,2,...,n$ where $d = w \times w \times ch$. Then, all image patches are normalized and whitened to remove correlations in the data [11]. Next, in the feature learning process, K-means algorithm is applied over the patch data to learn a dictionary $\mathbf{D}_{d \times K}$ which is composed of K bases (i.e., clustering centroids) $\mathbf{D}^{(j)} \in \mathbb{R}^d$.



(a) (b) (c)

Fig. 3: Feature training, encoding and feature selection on Fig. 2b. (a) Learned dictionary with $K = 200$ bases. (b) A comparison of encoded maps using Eq. (3) and our proposed Eq. (4). All maps are normalized to $[0,1]$. (c) Sorted dictionary bases according to feature map scores using Eq. (5).

Afterwards, in the encoding stage, distance from the i-th patch to the j-th base is calculated as $z_{ij} = \|\mathbf{x}^{(i)} - \mathbf{D}^{(j)}\|_2$. In [11], the following non-linear mapping function is used

$$f(z_{ij}) = max\big(0,\ mean(\mathbf{z}_{i\cdot}) - z_{ij}\big) \tag{3}$$

where $\mathbf{z}_{i\cdot} = [z_{i1}, z_{i2}, ..., z_{iK}] \in \mathbb{R}^K$ stores the distances from one patch $\mathbf{x}^{(i)}$ to all bases in the dictionary. However, when applied to our road detection task, the resulting maps of Eq. (3) lack spatial discrimination between road and non-road areas as is shown in the left column of Fig. 3b. This is because Eq. (3) sets nearly half responses to zero in each $\mathbf{z}_{i\cdot}$, while it does not consider the difference between road and non-road pixels in the same response map $\mathbf{z}_{\cdot j}$. To overcome this drawback, we propose to use the following encoding function

$$f(z_{ij}) = max\big(0,\ mean(\mathbf{z}_{\cdot j}) - z_{ij}\big) \tag{4}$$

where $\mathbf{z}_{\cdot j} = [z_{1j}, z_{2j}, ..., z_{nj}] \in \mathbb{R}^n$ is the distance from all patches to one base in the dictionary.[1] As is demonstrated in the right column of Fig. 3b, Eq. (4) assigns zero to about half areas of each feature map, which provides better spatial contrast between road and non-road areas than Eq. (3).

After feature mapping with Eq. (4), each feature map $\mathbf{z}_{\cdot j}$ is linearly normalized to range $[0,1]$. As can be observed from Fig. 3b, each base gives its highest responses to different areas, e.g., the four selected bases tend to represent road, bushes, sky, or edge. Based on this observation, it is preferred to select the feature maps which give high responses to the road areas and

[1] $\mathbf{z}_{\cdot j}$ also refers to a 2-D feature map in the following.

discard those bases that mainly respond to non-road regions. We use the following function to score each feature map

$$score(\mathbf{z}_{\cdot j}) = \sum_{i=1}^{n} \frac{\big(row(z_{ij}) - Rows/2\big)}{Rows/2} \cdot z_{ij} \tag{5}$$

where $row(\cdot)$ is the row position of z_{ij} in feature map $\mathbf{z}_{\cdot j}$ (from top to bottom), and $Rows$ is the total rows of the map. The coefficient $\big(row(z_{ij}) - Rows/2\big)/(Rows/2)$ assigns a weight to each mapped pixel under a linear range $(-1, 1]$, which is based on the assumption that road areas are more likely to appear in the lower part of the image. As a result, feature maps which have larger bottom areas with higher responses will receive higher scores. Then, the top m feature maps with the highest scores are selected (e.g., the upper-right map in Fig. 3b). Each selected feature map is denoted as $\hat{\mathbf{z}}_{\cdot j} \in \mathbb{R}^n, j = 1, 2, ..., m$, and the selected feature vector at each pixel is $\hat{\mathbf{z}}_{i\cdot} \in \mathbb{R}^m, i = 1, 2, ..., n$.

C. Hierarchical Threshold Segmentation

After feature learning, we aim to extract road areas using the selected feature maps. Although the features have been scored and filtered, there are always non-road responses in the selected maps. In addition to these non-road distractors, in some cases, the road surface is separated by lanes (e.g., the 11-th image in Fig. 5) which makes it difficult to determine road regions. So, in order to detect road areas automatically, we propose a multi-threshold segmentation algorithm, in which thresholds are determined based on the assumption on road positions and road feature reconstruction errors.

In [13], an unsupervised framework for optimal cluster extraction is designed for clustering algorithms. A tree-like hierarchical structure is first generated by adjusting some parameter, where each node represents an unsplit cluster in the feature space. Then, on the path from any leaf to the root of the tree, the node with the optimal "stability" is selected as a final cluster. In this paper, we extend the framework of [13] to segmentation tasks with two major variations. In particular, a tree structure is constructed in image's spatial space instead of pixels' feature space, and a discrete threshold parameter rather than a continuous variable is used to control nodes' generation and split. Moreover, we enhance the assessment of each node's stability by incorporating its pixels' position and their reconstruction errors, and find the road segmentation with a two-loop optimization process.

First of all, the average map of the selected features, i.e., $\bar{\mathbf{z}} = [\bar{z}_1, \bar{z}_2, ..., \bar{z}_n]$, is segmented by L thresholds to construct a tree, where $\bar{z}_i = mean(\hat{\mathbf{z}}_{i\cdot})$ and $\bar{z}_i \in [0, 1]$. In detail, first, L evenly spaced thresholds, i.e., $0/L, 1/L, ..., (L-1)/L$, are used to segment the map $\bar{\mathbf{z}}$. At each level, all connected components (4-connected) with no less than $minPts$ pixels are found by breadth-first search. Let R_{lr} denotes the r-th connected region at l-th level ($l = 1, 2, ..., L$). Then, a tree is constructed by parsing from higher to lower levels. Briefly, for a higher region $R_{(l-1)r}$ at level $l-1$, we find the lower regions $\{R_{ls}\}$ in level l which are covered by $R_{(l-1)r}$. If at least two lower regions exist (i.e., $|\{R_{ls}\}| \geq 2$), we set up new nodes for each regions. Otherwise, we put the only region into the same node as the higher one (i.e., $|\{R_{ls}\}| = 1$), or stop parsing this region (i.e., $|\{R_{ls}\}| = 0$). After tree construction, each tree node V_i contains regions from one or more levels and has one region in each level, i.e., $V_i = \{R_{(l)r_0}, R_{(l+1)r_1}, R_{(l+2)r_2}, ...\}$. An example of tree-construction from a feature map is illustrated in Fig. 4a.

After setting up the tree, we define the stability value to evaluate each node. For each pixel $\bar{z}_i$ in the average feature map, its stability is defined as

$$\begin{aligned} S(\bar{z}_i, \hat{\mathbf{z}}_{i\cdot}) = & \frac{row(\bar{z}_i) - Rows/2}{Rows/2} \\ & + \frac{-\big|col(\bar{z}_i) - Cols/2\big| + Rows/2}{Rows/2} \\ & + (-1) \cdot \lambda \cdot normalize\big(e_i\big) \end{aligned} \tag{6}$$

where $\hat{\mathbf{z}}_{i\cdot}$ is the feature vector corresponding to $\bar{z}_i$ and is involved in e_i's computation; $row(\cdot)$ and $col(\cdot)$ give the row and column position of $\bar{z}_i$ in the map; $Rows$ and $Cols$ are the sizes of the map; and $normalize(\cdot)$ adjusts all $e_i, i = 1, 2, ..., n$ into range $[0, 1]$ linearly. The first two items of Eq. (6) are locational weights on each pixel in accordance with our assumption that roads should locate in the bottom and horizontal middle of the image. The third item of Eq. (6) is an error between $\hat{\mathbf{z}}_{i\cdot}$ and its reconstructed version based on the road feature distribution which will be discussed below, and λ is a regularizer that trades off between pixel's location and reconstruction error. According to Eq. (6), the stability of $\bar{z}_i$ should be positive if $\bar{z}_i$ belongs to road (thus be vertically lower and horizontally centered in the image, and with small reconstruction error); otherwise, it should be negative. With Eq. (6), the stability for a region is defined as

$$S(R_{lr}) = \sum S(\bar{z}_i, \hat{\mathbf{z}}_{i\cdot}), \quad \forall \bar{z}_i \in R_{lr} \tag{7}$$

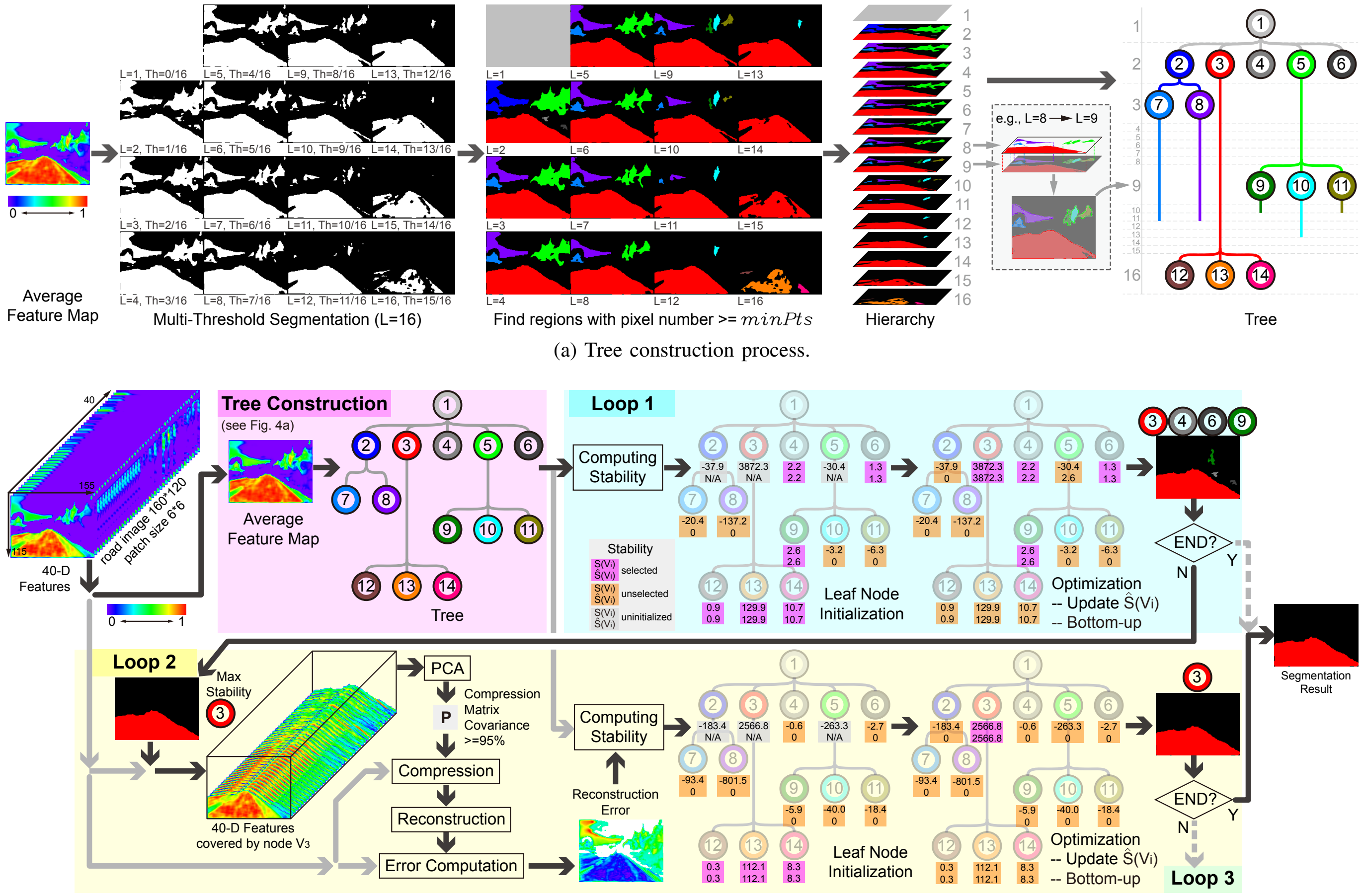
and further, the stability for a node is defined as

$$S(V_i) = \sum S(R_{lr}), \quad \forall R_{lr} \in V_i \tag{8}$$

With the definition of node's stability, the optimal nodes are selected as road segmentations according to the following object function

$$\begin{aligned} \max_{\delta_i} \quad & \sum_i \delta_i S(V_i) \\ s.t. \quad & \delta_i \in \{0, 1\} \\ & \sum_{j \in I_h} \delta_j \leq 1, \; \forall h \in Leaf \end{aligned} \tag{9}$$

where δ_i defines whether V_i is selected ($\delta_i = 1$) or not ($\delta_i = 0$), $Leaf$ stores all leaf nodes' indexes of the tree, and $I_h = \{j | V_j \text{ is ascendant of } V_h \text{ except } V_1\}$ includes all nodes on the path from a leaf V_h to the root V_1 but except V_1. The second constraint in Eq. (9) means that one or zero node is selected on each path from a leaf to the root, since no nodes are needed when all of them have stabilities below zero. This is different from [13] which requires exact one node



(a) Tree construction process.

(b) Algorithmic framework of hierarchical threshold segmentation, where $Iteration = 2$.

Fig. 4: Hierarchical threshold segmentation.

being selected on each path. Finally, the optimal nodes can be found by a bottom-up tree-parsing algorithm as follows.

```
function OPTIMIZE(·)
    for $\forall h \in Leaf$ do                         ▷ initialize all leaf nodes
        if $S(V_h) > 0$ then
            $\delta_h := 1$
            $\hat{S}(V_h) := S(V_h)$
        else
            $\delta_h := 0$
            $\hat{S}(V_h) := 0$
    for $\forall i \in (\{V_i\} - Leaf - V_1)$ do          ▷ bottom-up
        if $S(V_i) > 0$ AND $S(V_i) \geq \sum \hat{S}(V_i.child)$ then
            $\delta_i := 1$
            $\hat{S}(V_i) := S(V_i)$
            $\delta_j := 0, \forall j \in \{j | V_j \text{ is descendant of } V_i\}$
                                    ▷ set all lower nodes unselected
        else
            $\delta_i := 0$
            $\hat{S}(V_i) := \sum_{child} \hat{S}(V_i.child)$
    return $\delta_i$, $\hat{S}(V_i)$
```

where $\hat{S}(V_i) \geq 0$ stores intermediate values, and $\delta_i = 1$ points to the selected nodes.

In Eq. (6), the reconstruction error at $\bar{z}_i$ is defined as

$$e_i = \|\hat{\mathbf{z}}_{i\cdot} - \mathbf{P}\mathbf{P}^T \hat{\mathbf{z}}_{i\cdot}\|_2 \tag{10}$$

where $\mathbf{P}$ is a compression matrix computed via PCA (Principal Component Analysis). Specifically, features from a predefined road area R_{road} are used to compute $\mathbf{P}$, and 95% of road data covariance are preserved. Then, for a non-road pixel, there should be a large distance (error) between its original feature $\hat{\mathbf{z}}_{i\cdot}$ and the reconstructed version $\mathbf{P}\mathbf{P}^T\hat{\mathbf{z}}_{i\cdot}$, while the error should be small for road pixels. This method is also known as one-class classifier [14], [15].

However, there exists a chicken-and-egg problem, in that we do not have an exact road region R_{road} in the beginning. So, we propose a multi-iteration optimization process to solve this problem. In the first iteration, we set $\mathbf{P} = \mathbf{I}$ (i.e., let $e_i = 0$). While in the following iteration(s), we set R_{road} as the area covered by the output node with maximum stability from the previous iteration. The optimization algorithm is shown as follows.

```
procedure ITERATIVEOPTIMINZATION
    for $it = 1$ to $Iteration$ do
        if $it == 1$ then
            $\mathbf{P} := \mathbf{I}$
        else                                              ▷ PCA
            $\mathbf{P} := \mathrm{PCA}(\{\hat{\mathbf{z}}_{i\cdot} \in R_{road}\})$
            $e_i = \|\hat{\mathbf{z}}_{i\cdot} - \mathbf{P}\mathbf{P}^T\hat{\mathbf{z}}_{i\cdot}\|_2,\ i = 1, 2, ..., n$
        Compute $S(V_i)$ via Eq. (6) - (8)
        $\delta_i, \hat{S}(V_i) :=$ OPTIMIZE$(\cdot)$
        $j := \arg\max_i \hat{S}(V_i)$                    ▷ select $V_j$ as road
        $l := \min l,\ \forall R_{lr} \in V_j$             ▷ highest level of $V_j$
        $R_{road} := R_{lr}$       ▷ only $R_{l1}$ exists at highest level
```

Generally, we set $Iteration = 2$. And the regions pointed by $\delta_i = 1$ belongs to road areas. Before exporting the result, we use a morphological dilate operation to fill the gaps between segmented regions caused by edge patterns (e.g., lanes, skyline, etc).

III. Experiments

A. Experimental Setting

We test our proposed road detection algorithm[2] on 5 datasets, i.e., (1) After-Rain [16] with 251 images, (2) Sunny-Shadows [16] with 754 images, (3) 280 road images captured in Nanjing by our own camera, (4) Kitti-Layout [17] with 323 labeled images from the Kitti dataset [18], and (5) Road Detection Evaluation of Kitti Benchmark [18] with totally 290 test images which are grouped into 3 categories based on road types, i.e., Urban Marked (UM), Urban Multiple Marked (UMM) and Urban Unmarked (UU). Particularly, no ground truth is available for the test set of the Kitti Benchmark, and our results are evaluated on a server. For the first three datasets, we resize images to 160 × 120 pixels, while for the latter two, we resize them to 397*120 pixels in accordance with the original aspect ratio. Some examples from these datasets are shown in Fig. 5.

To assess the detection performance on dataset (1) to (4), we use the following measures [19]

$$P = \frac{\sum_i (A_i \bigcap M_i)}{\sum_i A_i}, R = \frac{\sum_i (A_i \bigcap M_i)}{\sum_i M_i}, F = \frac{2PR}{P+R} \quad (11)$$

where P, R and F are precision, recall and F-score on one dataset respectively. A_i, M_i and $A_i \bigcap M_i$ denotes the number of predicted pixels, ground-truth pixels and their intersections respectively in the i-th image. For the Kitti Benchmark, we adopt the maxF and AP values [18] reported by the server which are evaluated under bird-eye-view.

We set the algorithm parameters for dataset (1) to (4) as follows. In the preprocessing stage, we set $\sigma = 1.5$ for blurring. In the feature learning and selection stage, $6 \times 6 \times 3$ image patches are extracted for training, $K = 200$ bases are obtained after 10 iterations, and 20% of the total features are selected (i.e., $m = 40$). In the road segmentation stage, a tree is constructed with $L = 16$ thresholds and $minPts = 50$, node stabilities are computed with $\lambda = 2$, and we output the result after 2 iterations. While for the Kitti Benchmark, we only adjust σ to 2.5 for blurring, and the rest remains unchanged.

TABLE I: Road Detection Results

Dataset	Method	$P(\%)$	$R(\%)$	$F(\%)$
After-Rain [16]	Kong et al.'s [3]	71.00	97.23	82.07
	Xia et al.'s [12]	76.17	98.07	85.74
	Our Method	93.56	96.95	**95.22**
Sunny-Shadows [16]	Kong et al.'s [3]	92.20	61.32	73.65
	Xia et al.'s [12]	72.02	97.56	82.87
	Our Method	90.49	94.23	**92.32**
Nanjing	Kong et al.'s [3]	93.49	65.62	77.11
	Xia et al.'s [12]	87.28	95.44	91.18
	Our Method	92.32	91.86	**92.09**
Kitti-Layout [17]	Kong et al.'s [3]	68.07	78.47	72.90
	Xia et al.'s [12] [2]	34.36	99.75	51.11
	Our Method	76.65	72.33	**74.43**

TABLE II: Results on KITTI Road Detection Benchmark [18]

Method	UM		UMM		UU		Total	
	maxF(%)	AP(%)	maxF(%)	AP(%)	maxF(%)	AP(%)	maxF(%)	AP(%)
CN [17]	73.69	76.68	86.21	84.40	72.25	66.61	79.02	78.80
CN24 [7]	**86.32**	**89.19**	-	-	-	-	-	-
Stixel [8]	85.33	72.14	**93.26**	**87.15**	**86.06**	**72.05**	**89.12**	**81.23**
Ours	69.85	65.22	82.63	83.60	62.15	55.93	71.75	68.97

B. Road Detection Result

In Table I, we list the road detection performance of our algorithm on dataset (1) to (4), comparing with a vanishing-point-based method [3] and our previous work [12]. Some detection examples are shown in Fig. 5.

As can be seen in Table I, our algorithm outperforms other methods. Specially, in the first three datasets whose image aspect ratios are $4 : 3$, the F-scores of our method are consistently larger than 0.9. Whereas for Kitti-Layout in where image aspect ratio is about $3.3 : 1$, even though this dataset contains much more complex road scenes which makes it challenging for unsupervised methods, our algorithm still achieves $F = 0.74$ which is a better and more practical result than other methods.

In Table II, we compare our method with 3 deep supervised learning models on the Kitti Benchmark. As can be seen, our algorithm is not as good as those multi-layer networks, since we only use shallow features. However, our method still has the advantage of being unsupervised, whereas other models rely on large-scale labeled data for model training.

C. Impact of Parameters

We also evaluate the impacts of two key parameters on road detection accuracy. We choose two datasets for demonstration, i.e., After-Rain and Kitti-Layout. During evaluation, other parameters are kept the same as in Section III-A.

Firstly, the impact of the base number K in unsupervised feature learning on F-score is shown in Fig. 6a. As can be seen, when changing K in $[108, 500]$, the fluctuations of detection accuracy on both datasets are less than 0.02. This means that the number of clustering centroids does not have an obvious influence on detection accuracy. It might be caused by the

[2] Codes: http://github.com/JunkangZhang/UFL-HS-RoadDetection.

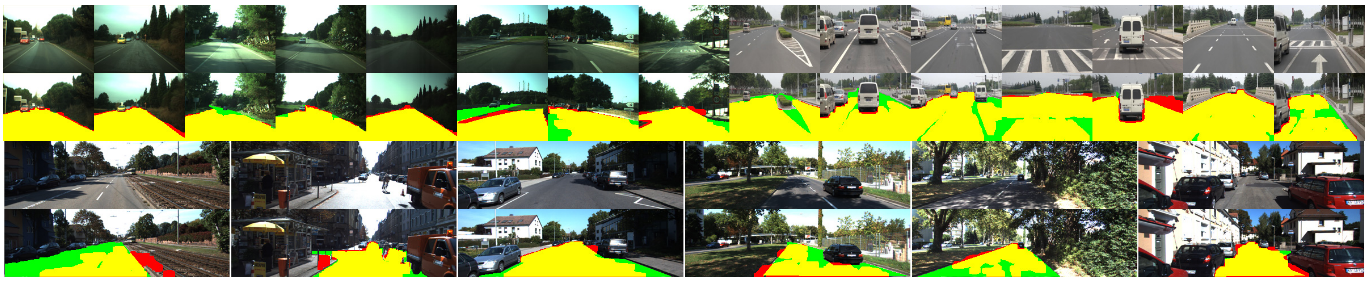
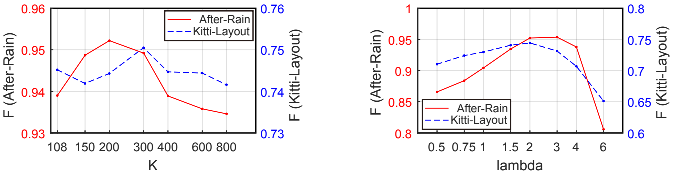


Fig. 5: Experimental results. Yellow and red stand for our predicted road areas, while yellow and green belong to ground-truth.



(a)

(b)

Fig. 6: F-score with different parameter settings. (a) K. (b) λ.

feature selection operation, since discriminative road features always exist and thus can be retained.

Besides, we evaluate the impact of the weight coefficient λ on the F-score. As can be seen in Fig. 6b, both datasets achieve high detection accuracy with $\lambda = 2$.

IV. Conclusion

In this paper, we developed an improved framework based on unsupervised feature learning for road detection from a single image. On the one hand, we proposed a new feature encoding method combined with a feature selection process to obtain discriminative road features. On the other hand, we designed a new segmentation algorithm to extract road areas from the learned feature maps. A tree is constructed to represent the regions' hierarchy via a multi-threshold segmentation, and a two-loop tree-parsing optimization is applied to find the most stable regions which is in accordance with the heuristic assumptions on the position of roads and small reconstruction error. Experimental results showed the effectiveness of our method on various benchmark datasets.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant 61671151, Natural Science Foundation of Jiangsu Province under Grant No.BK20130639, BK20131296, and BK20140649, Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

[1] J. M. Alvarez, T. Gevers, and A. Lopez, "Learning photometric invariance from diversified color model ensembles," in *Computer Vision and Pattern Recognition, IEEE Conference on*, 2009, pp. 565–572.

[2] J. M. Alvarez, T. Gevers, and A. M. Lopez, "Evaluating color representations for on-line road detection," in *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on*, 2013, pp. 594–599.

[3] H. Kong, J. Y. Audibert, and J. Ponce, "General road detection from a single image," *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 2211–2220, 2010.

[4] P. Moghadam, J. A. Starzyk, and W. S. Wijesoma, "Fast vanishing-point detection in unstructured environments," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 425–430, 2012.

[5] Z. He, T. Wu, Z. Xiao, and H. He, "Robust road detection from a single image using road shape prior," in *IEEE International Conference on Image Processing*, 2013, pp. 2757–2761.

[6] J. M. lvarez, A. M. Lpez, T. Gevers, and F. Lumbreras, "Combining priors, appearance, and context for road detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 3, pp. 1168–1178, 2014.

[7] Clemens-Alexander Brust, Sven Sickert, Marcel Simon, Erik Rodner, and Joachim Denzler, "Convolutional patch networks with spatial prior for road detection and urban scene understanding," in *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2015.

[8] Dan Levi, Noa Garnett, and Ethan Fetaya, "Stixelnet: A deep convolutional network for obstacle detection and road segmentation," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2015, pp. 109.1–109.12.

[9] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, 2004, vol. 1, pp. 1–2.

[10] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2006, vol. 2, pp. 2169–2178.

[11] Adam Coates, Andrew Y Ng, and Honglak Lee, "An analysis of single-layer networks in unsupervised feature learning," in *International conference on artificial intelligence and statistics*, 2011, pp. 215–223.

[12] S. Xia, J. Zhang, K. Lu, and K. Qin, "Road detection via unsupervised feature learning," in *IVCNZ*, 2015.

[13] Ricardo J. G. B. Campello, Davoud Moulavi, Arthur Zimek, and Jörg Sander, "Hierarchical density estimates for data clustering, visualization, and outlier detection," *ACM Trans. Knowl. Discov. Data*, vol. 10, no. 1, pp. 5:1–5:51, 2015.

[14] David Martinus Johannes Tax, *One-class classification*, TU Delft, Delft University of Technology, 2001.

[15] Jose M Alvarez, Theo Gevers, and Antonio M López, "Road detection by one-class color classification: Dataset and experiments," *arXiv preprint arXiv:1412.3506*, 2014.

[16] J. M. Alvarez and A. M. Lopez, "Road detection based on illuminant invariance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 184–193, 2011.

[17] Jose M Alvarez, Theo Gevers, Yann LeCun, and Antonio M Lopez, "Road scene segmentation from a single image," in *ECCV*, pp. 376–389. 2012.

[18] J. Fritsch, T. Khnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2013, pp. 1693–1700.

[19] J. Yuan, S. Tang, F. Wang, and H. Zhang, "A robust road segmentation method based on graph cut with learnable neighboring link weights," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 1644–1649.