

Robust Road Region Extraction in Video Under Various Illumination and Weather Conditions

Hadi Ghahremannezhad

Computer Science Department
New Jersey Institute of Technology
Newark, New Jersey 07102
Email: hg255@njit.edu

Hang Shi

Computer Science Department
New Jersey Institute of Technology
Newark, New Jersey 07102
Email: hs328@njit.edu

Chengjun Liu

Computer Science Department
New Jersey Institute of Technology
Newark, New Jersey 07102
Email: cliu@njit.edu

Abstract—Robust road region extraction plays a crucial role in many computer vision applications, such as automated driving and traffic video analytics. Various weather and illumination conditions like snow, fog, dawn, daytime, and nighttime often pose serious challenges to automated road region detection. This paper presents a new real-time road recognition method that is able to accurately extract the road region in traffic videos under adverse weather and illumination conditions. Specifically, the novel global foreground modeling (GFM) method is first applied to subtract the ever-changing background in the traffic video frames and robustly detect the moving vehicles which are assumed to drive in the road region. The initial road samples are then obtained from the subtracted background model in the location of the moving vehicles. The integrated features extracted from both the grayscale and the RGB and HSV color spaces are further applied to construct a probability map based on the standardized Euclidean distance between the feature vectors. Finally, the robust road mask is derived by integrating the initially estimated road region and the regions located by the flood-fill algorithm. Experimental results using a dataset of real traffic videos demonstrate the feasibility of the proposed method for automated road recognition in real-time.

I. INTRODUCTION

Road region extraction is a fundamental step in many modern computer vision applications, such as automatic driving, traffic warning, navigation, traffic surveillance, and driver assistance systems. Many studies have addressed the problem of vision-based road detection in recent years, in both applications of in-vehicle perception [3], [13], [23] and traffic video surveillance [9], [10], [14], [26]. The methods proposed in these studies are mostly applicable in both areas with some differences in the main motivations. Road detection helps with automatic driving, navigational warning and obstacle avoidance in the first group of applications and it is useful for region of interest (RoI) determination and traffic incidents detection in the second group.

The recent studies in road detection and RoI determination choose different strategies to segment the road region in the images. In some studies, the local features such as color [16], [17], brightness [31], texture [32], [34], or a combination of them [11], [33] are extracted in order to classify the pixels into road and non-road classes. Some methods tend to rely on the road models in order to match them with low-level features and detect the road region [6], [12], [18], [29]. Several

techniques suggest utilizing motion information and temporal features obtained from a sequence of video frames in order to extract the road area [14], [30]. Recently, convolutional deep neural networks have also been applied to segment the road region due to their ability in modeling non-linear variable relationships [1], [2], [5], [15], [19], [22]. In terms of road detection in traffic video analytic applications, the performance of supervised methods can suffer from a wide range of different illumination and weather conditions, image resolutions, camera's viewing angle, and distance from the road surface.

This study is focused on automatic road region extraction in traffic videos that aids with RoI determination which in turn reduces the need for computational resources and can be useful in automated detection of traffic incidents and driving violations. The use of a color features combined with gradient information and temporal features makes this method robust against illumination changes and severe weather conditions. The remainder of this paper is organized as follows. Section II describes the proposed road recognition method in details. In section II-A a method for initial road recognition based on color differences is introduced. Section II-B contains details on refining the extracted road region by using temporal and color features. The performance of the proposed method is evaluated in section III on real traffic videos, and the paper is concluded in section IV.

II. A FEATURE-BASED STATISTICAL METHOD FOR ROAD REGION RECOGNITION

Determining the region of interest (RoI) is crucial pre-processing step in many video analytic applications. The focus of this study is on finding the RoI in traffic videos, which is associated with the roadway region, in real-time and with no manual input. In this section, the main steps of the proposed method are discussed in details, which has mainly two contributions: (i) The new sampling approach can approximate the roadway location during the initial video frames solely based on color and temporal features without any assumption about the structure of the roadway. (ii) The initially estimated road is refined during an updating process that makes it robust against challenging illumination conditions.



Fig. 1: Sampling road regions from the background image based on the direction of moving vehicles. The red color indicated the sampled road pixels.

A. Initial road region approximation

In the tasks of traffic videos analysis, the RoI is associated with the roadway. In order to detect the location of the moving vehicles, a foreground segmentation method [27], [28] is applied. The road samples are taken from the subtracted background image in the corresponding location of the moving vehicles. An effective blob-tracking method [4] is utilized in order to estimate the moving direction of each vehicle. The foreground mask of each vehicle is then cropped based on its moving direction in order to filter out the potential non-road pixels. Figure 1 shows examples of the road sampling approach that is applied to construct a aggregated set of road samples Ω_{sm} .

The sampled road pixels are used in order to generate a road probability map based on color differences. According to the histogram models of the road samples, an example of which can be seen in Figure 2, the blue, green, and grayscale values of each pixel are discriminative features for classifying the road and non-road regions. A set of four-dimensional feature vectors is denoted by:

$$\mathcal{F}^t = \{f_1^t, f_2^t, \dots, f_N^t\} = \{f_i^t\}_{i=1}^N \quad (1)$$

where f_i^t is a D dimensional feature vector of pixel i that contains the blue, green, grayscale, and hue values of that pixel at frame t . The standardized Euclidean distance between each feature vector and the mean value of the pixels in the road sample are calculated as follows:

$$\begin{aligned} \bar{f}_j^t &= \left(\sum_{i \in \Omega_{sm}} f_{ij}^t \right) / |\Omega_{sm}^t|, j = 1 \dots D \\ (\sigma_j^2)^t &= \left(\sum_{i \in \Omega_{sm}} (f_{ij}^t - \bar{f}_j^t)^2 \right) / |\Omega_{sm}^t|, j = 1 \dots D \\ d_i^t &= \sqrt{\sum_{j=1}^D \frac{1}{(\sigma_j^2)^t} (f_{ij}^t - \bar{f}_j^t)^2} \end{aligned} \quad (2)$$

where $i = 1 \dots N$ is the pixel index, $D = 4$ is the number of features, $\bar{f}_j^t$ is the mean value of the j -th feature in the set of road samples Ω_{sm} , f_{ij}^t is the j -th feature of pixel i , d_i^t is the standardized Euclidean distance, $(\sigma_j^2)^t$ is the variance of the j -th feature at frame t . The road probability map $\mathcal{P}_{\mathcal{R}}$ is denoted by:

$$\mathcal{P}_{\mathcal{R}}^t = \{p_1^t, p_2^t, \dots, p_N^t\} = \{p_i^t\}_{i=1}^N \quad (3)$$



Fig. 2: The histogram plot representing the RGB and gray values of the background image from the road samples.

where N is the total number of pixels, and $p_i \in [0, 1]$ is the road probability value of pixel i at frame t which is calculated as follows:

$$\begin{aligned} \sigma_j^t &= \sqrt{\left(\sum_{i \in \Omega_{sm}} (f_{ij}^t - \bar{f}_j^t)^2 \right) / |\Omega_{sm}^t|}, j = 1 \dots D \\ \bar{\sigma}^t &= \left(\sum_{j=1}^D \sigma_j^t \right) / D \\ \lambda_i^t &= \max \left(0, \text{sgn} \left(d_i^t - \bar{\sigma}^t \right) \right) \\ p_i^t &= 1 - \lambda_i^t \left(\frac{d_i^t}{\sigma^t} + \frac{1}{k^2} \right), k-1 \leq \frac{d_i^t}{\sigma^t} < k \end{aligned} \quad (4)$$

where $i = 1 \dots N$ is the pixel index, $\bar{\sigma}^t$ is the mean standard deviation of the features among the D dimensions in Ω_{sm} , k is a natural number in $\{k \in \mathbb{N} | 1 < k \leq \max(d_i^t - \bar{\sigma}^t)\}$, and p_i is the resulting probability value of pixel i at frame t . Figure 3 represents an example of the extracted probability map from the difference values.

The extracted probability map $\mathcal{P}_{\mathcal{R}}$ is updated throughout the video frames by applying the temporal fusing algorithm as follows:

$$\begin{aligned} \hat{p}_i^t &= \frac{\sum_{f=1}^t w^f p_i^f}{1 + \sum_{f=1}^t w^f} \\ w^f &= |\Omega_{sm}^f| \end{aligned} \quad (5)$$

where $i = 1 \dots N$ is the pixel index, w^f is the weight of frame f which is associated with the number of pixels in the aggregated sample mask, p_i^f is the probability value of pixel i at frame f , Ω_{sm}^f is the accumulative road sample mask at frame f , N is the total number of pixels in each frame, and $\hat{p}_i^t$ is the updated probability value of pixel i . The Otsu's threshold [8] is applied on the resulting map in order to obtain a binary mask $\mathcal{P}_{\mathcal{R}}^*$.

B. Integration of temporal features with the estimated road region

Since road is a unified object, a combination of color and temporal features represent a more reliable estimation when enough time has passed. Here, the flood-fill algorithm is applied in order to unify the connected components of the road pixels and extract the road region. In order to define the limiting boundaries of the flood-fill method, the Canny edge detection method is applied on the difference image

Algorithm 1: Acquiring the accumulative foreground mask**Input:**

The size of each video frame
 The set T of vehicle tracks in the current frame
 The set of blobs for each track $B_t = \{b_1, \dots, b_n\}$
 A set of predefined thresholds $\mathcal{T} = \{\tau_d, \tau_i, \tau_s\}$

Output:

The accumulative foreground mask $\mathcal{F}_{acc}$ of the same size as the video frame

```

1 initialize  $\mathcal{F}_{acc}$  with 0;
2 foreach  $t \in T$  do
3   if  $size(t) < \tau_s$  then
4     continue;
5   end
6    $d = \|t_{cn} - t_{c1}\|$ ;
7   if  $d < \tau_d$  then
8     continue;
9   end
10  add track's current blob  $b_n$  to track's accumulative
    mask  $\mathcal{F}_t$ ;
11  if  $t_i > \tau_i$  then
12     $\mathcal{F}_{acc}[\mathcal{F}_t] = \mathcal{F}_{acc}[\mathcal{F}_t] + d$ ;
13  end
14 end
15  $\mathcal{F}_{acc} = \frac{\mathcal{F}_{acc}}{max(\mathcal{F}_{acc})}$ ;
```

$\mathcal{B}_d$ with $\tau_l = 0.66 \times M$ and $\tau_h = 1.33 \times M$ as the lower and upper thresholds, respectively, where M is the median luminance of $\mathcal{B}_d$. After applying the edge detection method, leak segmentation error can still occur due to lack of enough gradient information at the dominant road boundaries, which can be corrected by using the accumulative foreground mask $\mathcal{F}_{acc}$. Algorithm 1 shows the steps of accumulating the foreground masks obtained by the GFM [27], [28] method with false positives and slow-moving object filtered out by applying two thresholds τ_d and τ_s at steps 3–8. The threshold τ_i is used to define how long a track has to be inactive before being removed. The accumulative foreground mask $\mathcal{F}_{acc}$ is added by d in the location of the track only after track t has been removed from the set T (step 11 of Algorithm 1). This way the tracks with larger movements contribute more to the estimated road region. At the end, $\mathcal{F}_{acc}$ is normalized as it is divided by the maximum value.

Similar to the approach used in [7], [20], the contours of $\mathcal{F}_{acc}$ are smoothed using a Gaussian kernel. The Gaussian coefficients are calculated as follows:

$$\begin{aligned} \sigma' &= \frac{1}{2}(c\sigma + 1) \\ \mathcal{M} &= 2(\text{sgn}(\sigma')(|\sigma'| + 0.5)) - 1 \\ g_i &= \alpha \exp\left(\frac{-(i - \frac{\mathcal{M}-1}{2})^2}{2\sigma'^2}\right), \sum_{i=0}^{\mathcal{M}-1} g_i = 1 \end{aligned} \quad (6)$$

where c is an integer constant, $\mathcal{M} \in \{2n + 1 : n \in \mathbb{Z}\}$

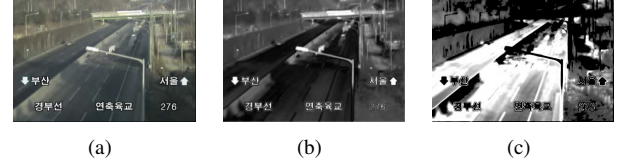


Fig. 3: Extracting the auxiliary road region probability map using feature difference values. (a) The subtracted background image $\mathcal{B}$. (b) The difference image $\mathcal{B}_d$ obtained by d_i . (c) The extracted probability map $\mathcal{P}_{\mathcal{R}}$.

is the Gaussian aperture size, σ is the standard deviation, α is the scale factor chosen so that $\sum_{i=0}^{\mathcal{M}-1} g_i = 1$, and g_i is the i -th Gaussian filter coefficient. The contours are smoothed separately over each X and Y axis:

$$\begin{aligned} C_j^k(n) &= \begin{cases} C_j(|C| + n - k) & , \text{if } n < k \\ C_j(n - k - |C|) & , \text{if } n > (k + |C| - 1) \\ C_j(n - k) & , \text{otherwise} \end{cases} \\ C_j^*(n) &= \sum_{i=0}^{\mathcal{M}-1} C_j^k(n) g_i, \quad k = -\mathcal{L} \dots \mathcal{L} \end{aligned} \quad (7)$$

where $n = 0 \dots (|C| - 1)$ is the index of each point on the curve, C is the surrounding contour of the accumulative foreground mask, $j \in \{x, y\}$ represents the x or y axis, $\mathcal{L} = \frac{1}{2}(\mathcal{M} - 1)$, and $C_j^*(n)$ is the position of the n -th point in the smoothed contour.

The sides of the smoothed contours which correspond to the boundaries of the road are partitioned into a set of K separate clusters $\mathcal{C} = \{c_k\}_{k=1}^K$ based on their connectivity which is in turn measured by Euclidean distance. The points of each cluster c_k are resampled by traversing in a pace equal to resample size $m_k = s_k/d$ where s_k is the arc-length of c_k and d is a pre-defined constant.

Then a similar approach to [21] is used to estimate the boundaries of the road by fitting a second-degree polynomial curve on each cluster. The principal component analysis (PCA) method is applied on each set of re-sampled points in order to calculate the direction of the maximum variation in the set. First a matrix $\mathbf{P}_k \in \mathbb{N}^{m_k \times 2}$ is formed with each row containing the x, y coordinate values of each resampled point from c_k . Then the covariance matrix $\mathbf{S}_k$ is computed as follows:

$$\begin{aligned} \mathbf{u}_k &= \frac{1}{m_k} \sum_{i=1}^{m_k} \mathbf{P}_k \\ \mathbf{S}_k &= \frac{1}{m_k - 1} \sum_{i=1}^{m_k} (\mathbf{P}_k - \mathbf{u}_k)(\mathbf{P}_k - \mathbf{u}_k)^T \end{aligned} \quad (8)$$

where $\mathbf{u}_k$ is a row vector that contains the mean $\bar{x}$ and $\bar{y}$ values of each column in $\mathbf{P}_k$. The eigenvalues and eigenvectors of

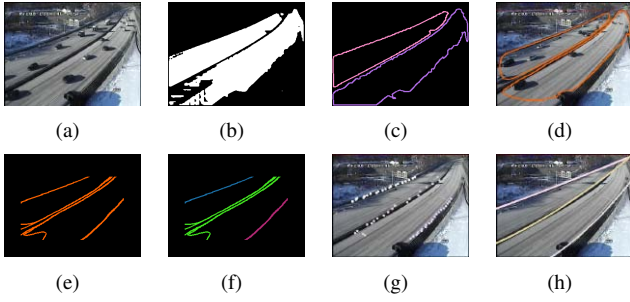


Fig. 4: Estimating the road boundaries. (a) Sample traffic video frame. (b) Accumulative foreground mask after one minute. (c) Contours of the accumulative foreground masks. (d) Smoothed contours. (e) Cropped contours. (f) Clustering. (g) Resampled points. (h) Estimated road boundaries.

the covariance matrix are calculated as follows:

$$\lambda_1^k, \lambda_2^k = \frac{1}{2} \left(\sigma_{x_k}^2 + \sigma_{y_k}^2 \pm \sqrt{(\sigma_{x_k}^2 - \sigma_{y_k}^2)^2 + 4\sigma_{x_k y_k}^2} \right) \quad (9)$$

$$\mathbf{e}_j^k = \frac{1}{\sqrt{\sigma_{x_k y_k}^2 + (\lambda_j - \sigma_{x_k}^2)^2}} \begin{bmatrix} \sigma_{x_k y_k}^2 \\ \lambda_j - \sigma_{x_k}^2 \end{bmatrix}$$

where $j \in \{1, 2\}$, $\sigma_{x_k}^2$, $\sigma_{y_k}^2$, and $\sigma_{x_k y_k}^2$ are the variance of x , variance of y , and covariance of xy values in $\mathbf{P}_k$, respectively. λ_j^k and $\mathbf{e}_j^k$ are the eigenvalues and their corresponding eigenvectors of $\mathbf{S}_k$. A matrix $\mathbf{E}_k$ is defined as follows:

$$\mathbf{E}_k = \begin{bmatrix} a_{11}^k & a_{12}^k \\ a_{21}^k & a_{22}^k \end{bmatrix} \quad (10)$$

where $\mathbf{e}_1^k = [a_{11}^k, a_{21}^k]^T$ and $\mathbf{e}_2^k = [a_{12}^k, a_{22}^k]^T$ are the first and second eigenvectors of $\mathbf{P}_k$, respectively. A new axis is generated and the data points from $\mathbf{P}_k$ are rotated as follows:

$$\theta_k = \cos^{-1}(\text{tr}(\mathbf{E}_k)/2)$$

$$\mathbf{R}_k = \begin{bmatrix} \cos \theta_k & -\sin \theta_k \\ \sin \theta_k & \cos \theta_k \end{bmatrix} \quad (11)$$

$$\mathbf{P}'_k = \mathbf{R}_k \mathbf{P}_k^T$$

where θ_k is the direction of maximum dispersion in $\mathbf{P}_k$, $\text{tr}(\mathbf{E}_k) = a_{11}^k + a_{22}^k$, $\mathbf{R}_k$ is the rotation matrix, and $\mathbf{P}'_k$ is the matrix containing the rotated points. After second-degree polynomial curve-fitting on each $\mathbf{P}'_k$, the resulting curves are rotated back to the original x and y axis to represent an estimation of the dominant road boundaries. Figures 4 and 5 represent an example of road boundary estimation.

The flood-fill algorithm with a connectivity value of 4 and with seed points taken from the set Ω_{sm} is applied in order to aggregate the connected components in a mask image $\mathcal{M}_{\mathcal{F}}$. During the component connection process, the maximal lower and upper thresholds of intensity difference between the current pixel and each of its nearest neighbors of the same component, or a new seed pixel being added to the

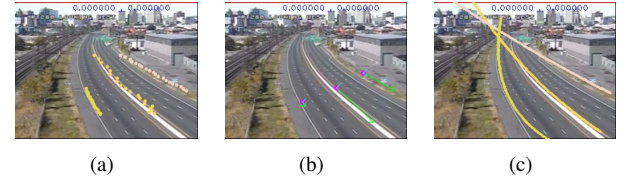


Fig. 5: Extracting the dominant road boundaries using the PCA method. (a) Re-sampled points used for curve fitting. (b) The direction of the maximum variation recognized by PCA. (c) The limiting boundaries estimated by curve fitting.

component are calculated based on the standard deviation of the background image $\mathcal{B}$ as follows:

$$\bar{B} = \frac{1}{N} \sum_{i=1}^N \mathcal{B}_i$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (\mathcal{B}_i - \bar{B})^2}{N}} \quad (12)$$

$$\tau = \max(1, \frac{\sigma}{k})$$

where $\bar{B}$ is the mean value of the background image $\mathcal{B}$, N is the total number of pixels in the background image, $\mathcal{B}_i$ is the intensity value of the i -th pixel, k is a pre-defined constant, and thr is the maximal lower or upper intensity difference.

When the intersection between the binary probability mask $\mathcal{P}_{\mathcal{R}}^*$ and the aggregated flood-fill mask $\mathcal{M}_{\mathcal{F}}$ surpasses a threshold, the flood-fill algorithm has connected most of the road components. Morphological procedure is performed on $\mathcal{M}_{\mathcal{F}}$ to bridge the gaps and the intersection between its result and $\mathcal{P}_{\mathcal{R}}^*$ is utilized as the final estimated road region as follows:

$$\mathcal{M}'_{\mathcal{F}} = \mathcal{M}_{\mathcal{F}} \oplus B$$

$$\mathcal{T} = \frac{|\mathcal{M}'_{\mathcal{F}} \cap \mathcal{P}_{\mathcal{R}}^*|}{|\mathcal{P}_{\mathcal{R}}^*|} \quad (13)$$

$$\mathcal{M}_{\mathcal{R}} = \begin{cases} \mathcal{P}_{\mathcal{R}}^* & , \text{if } \mathcal{T} < \theta \\ \mathcal{M}'_{\mathcal{F}} \cap \mathcal{P}_{\mathcal{R}}^* & , \text{otherwise} \end{cases}$$

where $\mathcal{M}'_{\mathcal{F}} = \{x | [(\hat{B})_x \cap \mathcal{M}_{\mathcal{F}}] \neq \emptyset\}$ is the result of a dilation operation with B as a structuring element, $\mathcal{T}$ is the number of common pixels between the probability mask and the accumulative flood-fill mask, $\theta \in [0, 1]$ is a predefined threshold, and $\mathcal{M}_{\mathcal{R}}$ is the final mask representing road pixels.

III. EXPERIMENTS

The performance of the proposed method is evaluated using several videos. The dataset is provided by the New Jersey Department of Transportation (NJDOT) which contains 84 real traffic videos captured from highways with different road structures, various illumination conditions, resolutions and frame-rates. A single frame from each tested video is shown at the first rows in Figure 6. The second rows represent the ground-truth mask corresponding to the road region in each video. The last rows illustrate the extracted road by a red-colored mask over the subtracted background of each video

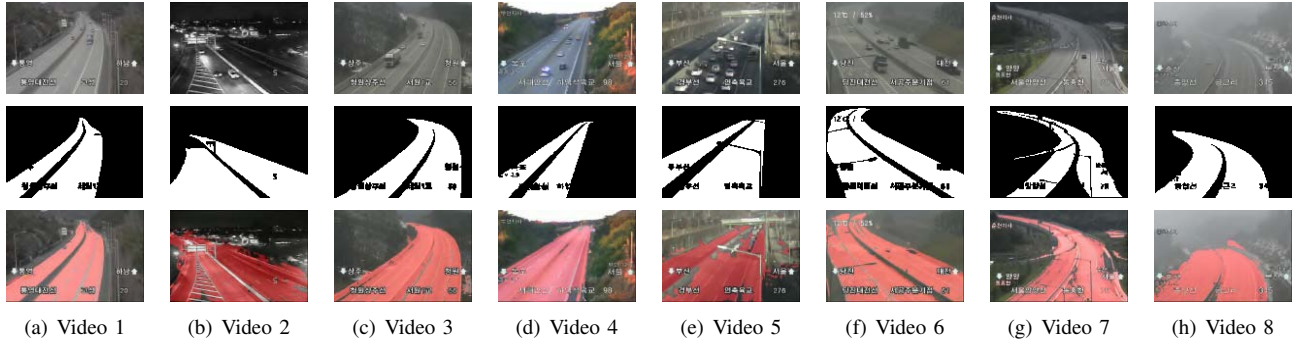


Fig. 6: Road extraction results in traffic videos with regular and challenging illumination conditions. The first row displays a sample frame of each video. The second row represents the ground-truth road region masks. The third row illustrates the extracted road region by the proposed method.

TABLE I: The quantitative evaluation of the proposed method

Video #	1	2	3	4	5	6	7	8	Average
Precision	0.98	0.87	0.94	0.93	0.89	0.97	0.80	0.97	0.93
Recall	0.96	0.93	0.95	0.94	0.96	0.89	0.92	0.81	0.92
F-Score	0.97	0.90	0.95	0.94	0.92	0.93	0.86	0.88	0.92

frame. All the experiments were conducted using a DELL XPS 8900 PC with a 3.4 GHz processor and 16 GB RAM. The average processing speed for video frames of size 720×480 pixels was around ~ 41.28 frames per second, which is inline with real-time requirements of video analysis applications. According to recent studies on embedded systems [24], [25], our proposed method can also be applied in embedded systems for real-time applications.

The following metrics are used in order to evaluate the quantitative results:

$$\begin{cases} FPR = F_P / (F_P + T_N) \\ PRE = T_P / (T_P + F_P) \\ REC = T_P / (T_P + F_N) \\ F_1 = 2 \times (PRE \times REC) / (PRE + REC) \\ ACC = (T_P + T_N) / (T_P + F_P + T_N + F_N) \end{cases} \quad (14)$$

where T_P , F_P are the number of pixels correctly and incorrectly reported as road regions, and T_N and F_N are the number of pixels that are correctly and incorrectly reported as non-road regions, respectively. FPR , PRE , REC , ACC , and F_1 refer to false positive rate, precision, recall, and F1-score, and accuracy, respectively. Figure 7 shows the changes in F1-score, accuracy, and the false-positive rate based on the frame number.

When the first vehicle is observed in each video, the initial road samples are obtained from its location and detection results improve when more vehicles pass along the road. In Table I the quantitative performance of the road region extraction method is reported for the 12 sample traffic videos. Due to under-segmentation, the entire roadway region is not always extracted which results in the precision values being slightly higher than the recall values in most cases. An

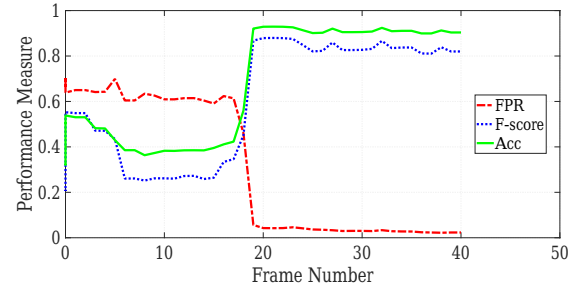


Fig. 7: The F-measure score, accuracy, and false-positive rate of the proposed method at different frames. The sudden improvement in the performance measures happens when the first vehicle is observed in the video sequence.

example of under-segmentation can be seen in Figure 6(h) which is mostly caused by the lose of tracking information at the far side of the road. Strong cast shadows and traffic congestion can also result in under-segmentation, specially at the initial frames (e.g., Figure 6(e)). Over-segmentation or leak segmentation can also occur which is usually due to the lack of sufficient gradient information at the road boundaries or the illumination effects which causes similarity between the non-road and road regions. Examples of over-segmentation are exemplified in Figures 6(b), 6(d) and 6(h).

IV. CONCLUSION

Determining the region of interest (RoI) in video analysis applications is an important pre-processing step. An accurate RoI can keep all the essential information while reducing the need for computational resources and eliminating a portion of potential false outputs. In applications of traffic video analysis the RoI usually refers to the road region. In this paper, an

adaptive and fully automatic method is proposed for road recognition in real-time. Temporal features are utilized in a statistical foreground segmentation method in order to obtain sample data. The road region is then extracted in the initial frames of the video by using pixel values in different color-spaces. The initially extracted road is updated and refined with more sample data generated through the subsequent video frames.

The proposed method shows good performance in videos of different sizes and frame-rates with various illumination and weather conditions. Since the first part of the proposed method uses feature-based probability estimations, it can be applied in applications with in-vehicle perception as well with the initial road samples gathered by a triangular area in front of the vehicle. This method is not limited to geometrical models since no assumptions have been made about the structure of the road. The extracted road region can further be utilized as the RoI in different traffic video analysis tasks. The experimental results from evaluating the method using real traffic videos provided by NJDOT indicate the feasibility of the proposed method for real-world applications.

ACKNOWLEDGMENT

This paper is partially supported by the NSF grant 1647170.

REFERENCES

- [1] N. Y. Q. Abderrahim, S. Abderrahim, and A. Rida, "Road segmentation using u-net architecture," in *2020 IEEE International conference of Moroccan Geomatics (Morgeo)*. IEEE, 2020, pp. 1–4.
- [2] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "Lidar-camera fusion for road detection using fully convolutional neural networks," *Robotics and Autonomous Systems*, vol. 111, pp. 125–131, 2019.
- [3] C.-K. Chang, J. Zhao, and L. Itti, "Deepvp: Deep learning for vanishing point detection on 1 million street view images," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.
- [4] F. Chang, C.-J. Chen, and C.-J. Lu, "A linear-time component-labeling algorithm using contour tracing technique," *Computer Vision and Image Understanding*, vol. 93, no. 2, pp. 206–220, 2004.
- [5] Z. Chen, J. Zhang, and D. Tao, "Progressive lidar adaptation for road detection," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 693–702, 2019.
- [6] G. Cheng, Y. Wang, Y. Qian, and J. H. Elder, "Geometry-guided adaptation for road segmentation," in *2020 17th Conference on Computer and Robot Vision (CRV)*. IEEE, 2020, pp. 46–53.
- [7] S. Elghoul and F. Ghorbel, "An efficient 2d curve matching algorithm under affine transformations," in *VISIGRAPP (4: VISAPP)*, 2018, pp. 474–480.
- [8] T. Y. Goh, S. N. Basah, H. Yazid, M. J. A. Safar, and F. S. A. Saad, "Performance analysis of image thresholding: Otsu technique," *Measurement*, vol. 114, pp. 298–307, 2018.
- [9] M. A. Helala, K. Q. Pu, and F. Z. Qureshi, "Road boundary detection in challenging scenarios," in *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*. IEEE, 2012, pp. 428–433.
- [10] M. A. Helala, F. Z. Qureshi, and K. Q. Pu, "Automatic parsing of lane and road boundaries in challenging traffic scenes," *Journal of electronic imaging*, vol. 24, no. 5, p. 053020, 2015.
- [11] T. Huang, Z. Wang, X. Dai, D. Huang, and H. Su, "Unstructured lane identification based on hough transform and improved region growing," in *2019 Chinese Control Conference (CCC)*. IEEE, 2019, pp. 7612–7617.
- [12] H. U. Khan, A. R. Ali, A. Hassan, A. Ali, W. Kazmi, and A. Zaheer, "Lane detection using lane boundary marker network with road geometry constraints," in *The IEEE Winter Conference on Applications of Computer Vision*, 2020, pp. 1834–1843.
- [13] T. Kim, Y.-W. Tai, and S.-E. Yoon, "Pca based computation of illumination-invariant space for road detection," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 632–640.
- [14] Q.-J. Kong, L. Zhou, G. Xiong, and F. Zhu, "Automatic road detection for highway surveillance using frequency-domain information," in *Proceedings of 2013 IEEE International Conference on Service Operations and Logistics, and Informatics*. IEEE, 2013, pp. 24–28.
- [15] J.-S. Lee and T.-H. Park, "Fast road detection by cnn-based camera-lidar fusion and spherical coordinate transformation," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [16] D. Liu, Y. Wang, and T. Chen, "Application of color filter adjustment and k-means clustering method in lane detection for self-driving cars," in *2019 Third IEEE International Conference on Robotic Computing (IRC)*. IEEE, 2019, pp. 153–158.
- [17] D. Liu, Y. Wang, T. Chen, and E. T. Matson, "Accurate lane detection for self-driving cars: An approach based on color filter adjustment and k-means clustering filter," *International Journal of Semantic Computing*, vol. 14, no. 01, pp. 153–168, 2020.
- [18] S. Luo, X. Zhang, J. Hu, and J. Xu, "Multiple lane detection via combining complementary structural constraints," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [19] Y. Lyu, L. Bai, and X. Huang, "Road segmentation using cnn and distributed lstm," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2019, pp. 1–5.
- [20] F. Mai, C. Chang, and Y. Hung, "Affine-invariant shape matching and recognition under partial occlusion," in *2010 IEEE International Conference on Image Processing*. IEEE, 2010, pp. 4605–4608.
- [21] N. Parvin, "Robust curved road boundary identification using hierarchical clustering," Ph.D. dissertation, 2013.
- [22] J. W. Perng, Y. W. Hsu, Y. Z. Yang, C. Y. Chen, and T. K. Yin, "Development of an embedded road boundary detection system based on deep learning," *Image and Vision Computing*, p. 103935, 2020.
- [23] T. Rateke and A. von Wangenheim, "Road surface detection and differentiation considering surface damages," *arXiv preprint arXiv:2006.13377*, 2020.
- [24] M. Reichenbach, L. Liebischer, S. Vaas, and D. Fey, "Comparison of lane detection algorithms for adas using embedded hardware architectures," in *2018 Conference on Design and Architectures for Signal and Image Processing (DASIP)*. IEEE, 2018, pp. 48–53.
- [25] A. Rossi, N. Ahmed, S. Salehin, T. H. Choudhury, and G. Sarowar, "Real-time lane detection and motion planning in raspberry pi and arduino for an autonomous vehicle prototype," *arXiv preprint arXiv:2009.09391*, 2020.
- [26] M. Santos, M. Linder, L. Schnitman, U. Nunes, and L. Oliveira, "Learning to segment roads for traffic analysis in urban images," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 527–532.
- [27] H. Shi and C. Liu, "A new foreground segmentation method for video analysis in different color spaces," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 2899–2904.
- [28] —, "A new global foreground modeling and local background modeling method for video analysis," in *International Conference on Machine Learning and Data Mining in Pattern Recognition*. Springer, 2018, pp. 49–63.
- [29] S. Srivastava and R. Maiti, "Multi-lane detection robust to complex illumination variations and noise sources," in *2019 1st International Conference on Electrical, Control and Instrumentation Engineering (ICECIE)*. IEEE, 2019, pp. 1–8.
- [30] L.-W. Tsai, Y.-C. Chean, C.-P. Ho, H.-Z. Gu, and S.-Y. Lee, "Multi-lane detection and road traffic congestion classification for intelligent transportation system," *Energy Procedia*, vol. 13, pp. 3174–3182, 2011.
- [31] J. Wang, T. Mei, B. Kong, and H. Wei, "An approach of lane detection based on inverse perspective mapping," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 35–38.
- [32] X. Wang, Y. Guo, J. Wang, and M. Song, "Self-made texture and clustering based road recognition for ugv," in *2017 36th Chinese Control Conference (CCC)*. IEEE, 2017, pp. 10999–11004.
- [33] L. Xiao, R. Wang, B. Dai, Y. Fang, D. Liu, and T. Wu, "Hybrid conditional random field based camera-lidar fusion for road detection," *Information Sciences*, vol. 432, pp. 543–558, 2018.
- [34] G. Yang, Y. Wang, J. Yang, and Z. Lu, "Fast and robust vanishing point detection using contourlet texture detector for unstructured road," *IEEE Access*, vol. 7, pp. 139 358–139 367, 2019.