Intelligent Vision-Based Real-Time Detection for Rough Terrain Navigation Robot

Rami A. AL-Jarrah
Department of Mechanical Engineering, Faculty of Engineering, The Hashemite University, Zarqa, Jordan
Email: ramia@hu.edu.jo

***Abstract*—The environmental perception is very important for autonomous intelligent ground robots in the outdoor areas. One of the essential tasks for these robots is to detect the unstructured regions autonomously with low computational complexity. Thus, this paper presents the modeling of a computer vision-based robot which is capable of adjusting itself into various structured and unstructured environments. To resolve the variety of road types, the learning algorithm architecture is designed to formulate the problem as a sequential road type classification. The classification is designed to contain two types: normal road and curved road. In normal road, the vanishing point estimated is proposed by using the artificial neural network. Then, the appearance model based upon multivariate Gaussian is quickly constructed from a sample region that is determined by the vanishing point and dominant left and right borders. The trapezoidal fuzzy membership functions is proposed to find the threshold value that depends on the orientation of pixel and given regions to ensure the accuracy of the segmentation. In curved road, the improved Fuzzy C-means algorithm which is fast superpixel algorithm based upon membership filtering is implemented. This provides the full capability of achieving better pre-segmentation results with significantly shorter runtime and more robustly results. The proposed approach is evaluated by using two datasets. Implementation the proposed method on the rover bogie robot has shown its ability in the real-time navigation.**

***Index Terms*—ConvNets, fuzzy logic, detection, road detection, vanishing point**

I. INTRODUCTION

Recently, the researches on autonomous navigation system have been keen many attentions and attraction. One essential issue of these methods is vision-based road detection in unstructured regions. The detection itself is classified as a binary problem which tries to classify each pixel on the input image as either a road or non-road category. Therefore, the road detection is still a challenging research not only because of road variations scene such as different colors and textures, but also the variations of several conditions like different illuminations and weather conditions. In [1], they introduced vision-based road detection for a robot's path following at off-road regions. However, the proposed approach based on a road region must satisfy the mechanical traversability and the far-field capability simultaneously. Some methods are based on color features [2], combines texture and color features [3], and road boundaries [4]. However, these approaches do not work well in the unstructured roads. In addition, several vision-based approaches have been proposed for structured and unstructured road detection [5-14]. Some approaches have been proposed to tackle the challenging unstructured road conditions to achieve a good balance between the accuracy and the time efficiency [5]. However, it is still challenging for mobile robots to navigate safely in real-time and autonomously locate variety of unstructured areas that has no painted markers or distinguishable borders [6]. Estimated the vanishing point and road segmentation was proposed in [7]. Even though this approach has ability to detect various types of roads, but it is very limited due to either the high computational complexity of the vanishing point or when the road has curved boundaries. Several researchers have proposed different models to speed up the estimation procedure of the vanishing point [8, 9]. In addition, many approaches have differentiated the road pixels from the image using the appearance models [10–11]. There are also several approaches based upon constant sample regions such as using illuminant invariance space [12] and selected a region as road masks to model road and background [13]. In [14], the author introduced initial trapezoidal region at bottom of the image and then update it during the process. However, the performance and the accuracy of these approaches based upon the quality of sample area that maybe a non-road sample region. The detection road regions in varying conditions based upon the geographical information and road geometry was presented in [15]. Also, the deep learning methods are implemented for region segmentation such as structured random forest [16] and efficient deep model [17]. However, the runtimes for their approaches were 70ms and 80ms, respectively [16, 17]. Therefore, all of the approaches mentioned above have high and expensive runtime and they are still very challenging to implement in a robot for real-time applications.

For this paper, a novel road classifier and detector based on multivariate Gaussian, trapezoidal fuzzy membership functions, FCM as well as the neural ConvNets is proposed.

In this paper, we present a novel approach for detecting traversable regions for a ground mobile robot from a single image in real-time without requiring any

Manuscript received April 6, 2021; revised July 12, 2021.

© 2021 Int. J. Mech. Eng. Rob. Res
doi: 10.18178/ijmerr.10.12.645-659

special hardware. The main contributions of the paper can be briefly described as follows:

1. The proposed method starts classifying the road type based on clustering features learning model into curved and non-curved roads. This classifying is very essential to come with the problem of segmentation and road edges problems. The road inference is responsible for infers result and determine road type.
2. A new proposed method is presented to robustly estimate the vanishing point (VP) based on Gabor filter bank and ConvNets which considers the problem as regression problem. The Gabor filter bank is used to extract image texture information in the preprocessing step, thereby enhancing the generalization. The ConvNets is implemented to predict the position of the VP. This estimator outperforms the state-of-the-art considering the accuracy and the tradeoff of time efficiency.
3. For the non-curved road type: A new fast and self-supervised segmentation scheme using trapezoidal fuzzy membership functions is proposed for unstructured traversable region detection. The texture orientation is computed using Gabor filter to create set of imaginary rays and to find out the right and left dominant borders. Then, the trapezoidal fuzzy membership functions will compute the threshold value that depends on the orientation of pixel and the given regions. This threshold value is very important to ensure the accuracy of the segmentation against the noise.
4. Adapted the FCM combined with trapezoidal fuzzy membership function is presented as a new fast and self-supervised segmentation to deal with challenged curved road type. This is because the method in (C) with dominant boarders is slightly affected when challenged curved borders presents. Therefore, the superpixel method which depends on Fuzzy C-mean algorithm based upon morphological reconstruction and membership filtering is used to provide better pre-segmentation results with shorter run time.

The remainder of the paper is organized as follows. The proposed method is described in details in Section II. Experimental results and discussion are presented in Section III. Finally, the conclusion is drawn in Section IV.

II. The Vision-based Road Detection Method

The proposed algorithm for the vision-based road detection is capable of working on either the structured roads or the non-structured roads surfaces. The general framework of the proposed road detection is shown in Fig. 1. The algorithm starts by classifying the road type based on clustering features learning model. This classifying is very essential to come with the problem of segmentation and road edges problems. Thus, the learning model is presented in order to classify the road type into two main categories: normal road or curved road. The 5000 images are used for training and testing the learning model. Then, the road inference is accountable for finding out the road type by infer the input image result. For normal road classification, the texture orientation is computed using Gabor filter with 36 directions for each pixel of input image. After creating set of imaginary rays and finding the right and left borders, the ConvNets will be implemented to estimate the vanishing point (VP). Moreover, the seed pixel point belongs to the road region is located through the constraints of VP and two road borders candidates. The sample region for the segmentation stage is surrounding the seed point and it is selected to model the road combining RGB and IIS features using multivariate Gaussian method. Then, the trapezoidal fuzzy membership functions is proposed to find the threshold value that depends on the orientation of pixel and given regions to ensure the accuracy of the segmentation against the noise. Moreover, the implemented road estimator with two boarders is not sufficient with curved borders. Hence, it is not possible to represent the border of the sample region from VP using the imaginary rays. Therefore, the superpixel Fuzzy C-mean algorithm based upon morphological reconstruction and membership filtering is proposed which is fast and robust and it is able to provide better pre-segmentation results with shorter run time with curved road type.

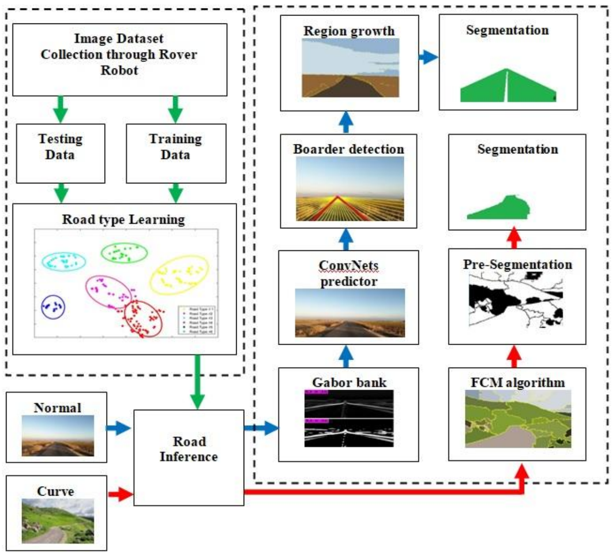


Figure 1. The framework for road detection.

A. Problem Definition and Road Type

Let $\mathbf{P}_{ex}= \{p_1, p_2 \ldots p_N\}$ is the pixels of input images from a training dataset and $\mathbf{LB} = \{LB_1, LB_2 \ldots LB_N\}$ is the label of the pixel where $\mathbf{LB = 1}$ indicates the pixel is in road pattern in the road region $\mathbf{R_R}$ and $\mathbf{LB= 0}$ indicates non-road pixel. Therefore, the road region $\mathbf{R_R}$ has to satisfy the following

$$\begin{cases} \forall p_i \in R_R & if.LB_i = 1 \\ \forall p_i \notin R_R & if.LB_i = 0 \end{cases} \tag{1}$$

The model of the road type is $\boldsymbol{\theta}_W= \{\theta_1, \theta_2, \theta_3, \theta_4 \ldots \theta_W\}$ and this model will be trained to infer the value. The problem of finding $\mathbf{R_R}$ is converted into the reasoning of a joint classification and segmentation: **Pr(LBt ,Wt|Pex,t ,Rt-1,θ)**

$$\Pr(LB_t, W_t \mid P_{ex,t}, R_{t-1}, \theta) = \Pr(W_t \mid P_{ex,t}, R_{t-1}, \theta) \cdot \Pr(LB_t \mid W_t, P_{ex,t}, R_{t-1}, \theta) \quad (2)$$

$$\Pr(LB, W \mid P_{ex}, R_{t-1}, \theta) = \Pr(W \mid P_{ex}, R_{t-1}, \theta) \cdot \Pr(LB \mid W, P_{ex}, R_{t-1}, \theta_W) \quad (3)$$

The online inference for road detection is solved consecutively by first obtaining the road type **Pr(W|Pex, Rt-1,θ)** then extracting road region **Pr(LB|W,Pex,Rt-1,θ)** with road model θ that is learned. While given a road region **Rt-1** and road model **θ**, the classification of the road in the current frame is solved by maximizing **Pr(W|Pex, Rt-1,θ)**. Therefore, we obtain the road type inference as follows:

$$\widehat{W} = \arg\max_W \Pr(W \mid P_{ex}, R_{t-1}, \theta) = \arg\max_W \prod_{P_{ex,i} \in R_t} \Pr(P_{ex,i} \mid \theta_W) \quad (4)$$

The evaluation of the pixels in the current region $\mathbf{R_t}$ has the priority in the process. Thus, comparing any features similarity between the $\mathbf{R_R}$ and the proposed road models by SURF algorithm is the introduced to measures the similarity distance function between current $\mathbf{R_t}$ and the models $\boldsymbol{\theta_W}$. Moreover, the Gaussian kernel in feature space has to be used in order to classify road's types. Therefore, assuming J road types, the training data T_{Data} will be described as $\{T_{Data,1}, T_{Data,2}, \ldots, T_{Data,m}\}$ and it will specify the features of the road. Hence, the T_{Data} could be expressed as distribution function g that uses the Gaussian mixture model (GMM) based upon the mixing weight π_J.

$$\pi_J = \sum_1^J \pi_J = 1 \quad \{0 \le \pi_J \le 1\} \quad (5)$$

$$g(T_{Data}; \mu, \textstyle\sum) = \frac{1}{\sqrt{(2\pi)^D \mid \sum \mid}} \exp\left[-0.5 (T_{Data} - \mu)^T \textstyle\sum^{-1} (T_{Data} - \mu)\right] \quad (6)$$

$$p(T_{Data} \mid \theta) = \sum_1^J \pi_J g_J(T_{Data}; \mu_J, \Sigma_J) \quad (7)$$

During learning process, $\theta = \{\theta_1, \theta_2, \theta_3, \theta_4 \ldots \theta_N\}$ has to be solved where θ_J is a triple $\{\pi_J, \mu_J, \sigma_J\}$. Thus, the log-likelihood function is given as

$$L(T_{Data} \mid \theta) = \sum_1^M \ln\left[\sum \pi_J g_J(T_{Data}; \mu_J, \Sigma_J)\right] \quad (8)$$

The expectation of this likelihood functoin is obtained after J+1iterations and the maximizing Q will perform the value of θ_{J+1}.

$$Q(\theta \mid \theta_J) = E\{L \mid T_{Data}, \theta_J\} \quad (9)$$

$$\theta_{J+1} = \arg\max Q(\theta \mid \theta_J) \quad (10)$$

The above two steps are repeated until convergence or the maximum number is reached. As an example of the clustering features for road type is shown in Fig. 2.

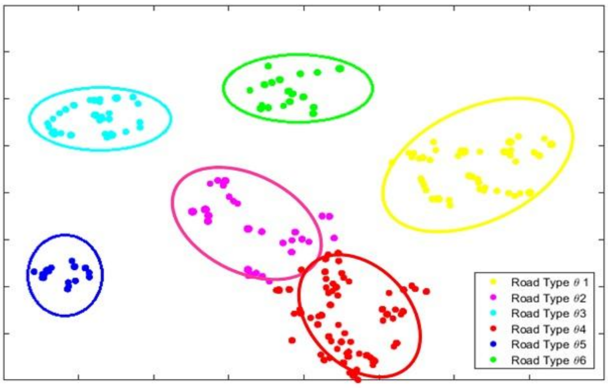


Figure 2. Road type based on clustering features

B. Gabor Filter Bank

Gabor Transform has been implemented because it is very robust to variable illumination and noise besides its ability to localize the extracted frequencies and provide orientations information.

$$\psi(x, y, \omega_0, f) = \frac{\omega_0}{\sqrt{2\pi K}} \left(e^{i\omega_0 (X')} - e^{\frac{-K^2}{2}} \right) . e^{\frac{-\omega^2}{8K^2}\left(4(X')^2 + Y'^2\right)} \quad (11)$$

where **f** is the orientation of the Gaussian envelope (Gabor Transform orientation), w is the radial frequency (the window size), **K** is the size of the Gabor filter. The orientation interval **[0- π]** was regularly sampled into **36** orientations with a step of **n* π/36 (n=0, 1, 2, …, 35)**.

In this research the Gabor filter bank consists of **180** filters (**5 scales × 36 orientations**). The gray image will convolve and the result is **36x5** response images with each image emphasizes edges of a given orientation in a given scale. The magnitude is computed to the best characterize the local texture properties $\mathbf{I_{w,f}(p)}$.

Therefore, the average of the Gabor magnitude is computed to get only one Gabor magnitude response $\mathbf{G_{w,f}(p)}$ per orientation. That means the response of different scales $\mathbf{R_f(p)}$ is averaged to reduce the number of images to 36 rather than **180** images as it is illustrated in Fig. 3.

$$I_{w,f}(p) = \sqrt{\left[\mathrm{Re}(G_{w,f}(p))\right]^2 + \left[\mathrm{Im}(G_{w,f}(p))\right]^2} \quad (12)$$

$$R_f(p) = mean(I_{w,f}(p)) \quad (13)$$

The pixel is brighter, the more likely the pixel is contained on an edge with orientation **f**. In order to obtain the binary images the threshold was applied to the 36 average responses. Moreover, at each point **p** the dominant texture orientation $\mathbf{f_{max}}$ that corresponds to the

© 2021 Int. J. Mech. Eng. Rob. Res

maximum average magnitude of the Gabor response is given by:

$$f_{\max}(p) = \max_f R_f(p) \tag{14}$$

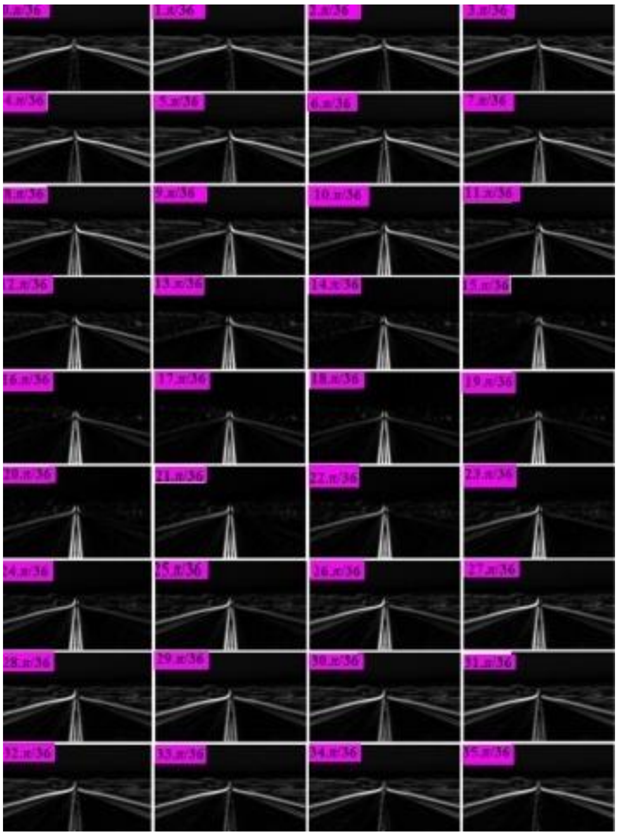


Figure 3. 36 corresponding average responses

C. Predictor Algorithm for Vanishing Point

The deep neural network learning methods such as Convolutional Neural Networks (ConvNets) have shown great success in computer vision because of its adaptability and it has no necessity of artificial modeling. Regardless that it is widely used in target detection, identification and vanishing point detection, but there is still few researches of vanishing point detection based upon the depth learning method. The ConvNets which consists of the convolution layer and pool layer can be implemented to detect the vanishing point in image. However, the pool layer is very useful for the position classification problem rather than position independent regression problem [18, 19]. Therefore, a simple ConvNets is proposed as it is illustrated in Fig. 4. The novel proposed ConvNets predictor, which considers the problem as regression problem, contains two main sections: the extraction of image texture information and the merging of the predictions. The final vanishing point is computed by the mean value of prediction at 36 magnitude responses.

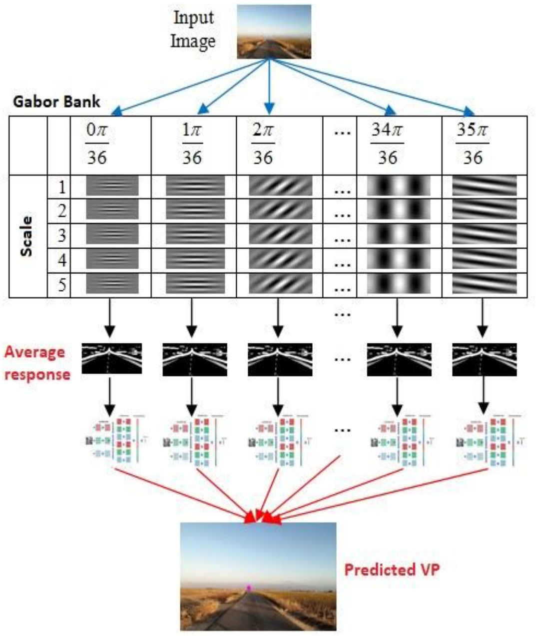


Figure 4. General layout diagram of the proposed VP detector

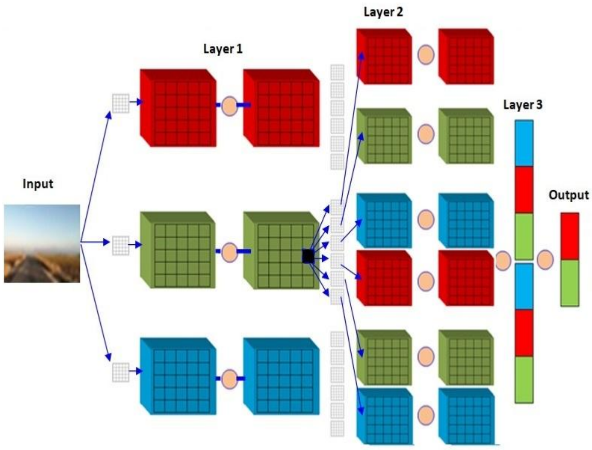


Figure 5. The architecture of convolutional neural network.

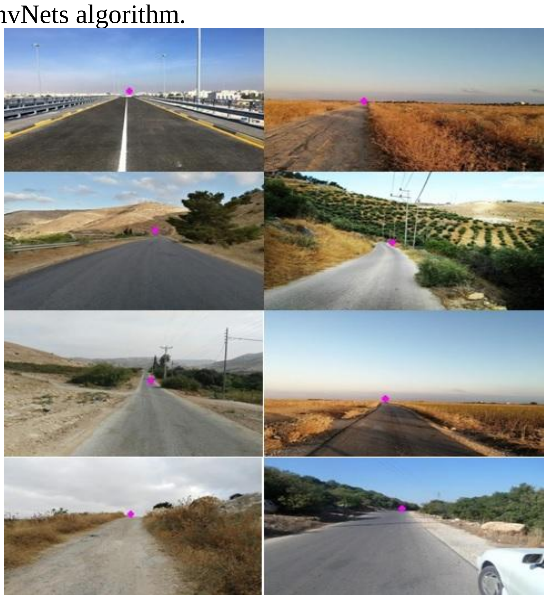
As it is shown in Fig. 5, the first hidden layer convolves 3 filters (5×5) with stride 2. When the stride is 2, the filters jump two pixels at a time during sliding around to produce smaller output volumes. The second hidden layer convolves 6 (5×5) filters with stride is 2. The third hidden layer is a full connection layer. Immediately after each layer, it is convention to apply the nonlinear Rectified Linear Units (ReLU) as the activation function by applying f(x) = max (0, x) thresholding at zero. This increases the nonlinear properties of the model and introduces nonlinearity to a system which has computed linear operations without affecting the receptive fields of the Convolutional layer. Also, the ReLU provides ability to the network to train faster with no significant difference to the accuracy. Because the gradient decreases exponentially through the lower layers, ReLU helps to alleviate vanishing gradient issue in where lower layers of network are slowly training. The output layer is a fully connected linear layer with 2 neurons that is obtained through the ratio of the vertical and horizontal coordinates of the VP in the input image. In order to get the training sample set, a total of 4000 images were used as Dataset 1. The total of 3000 images was randomly selected as the training sample and the VP location was manually fed in each image during the training process. The remaining images were used as the test sample set. After filtered by the Gabor filter bank, the texture feature frames of five scales and 36 directions were obtained as shown in Figure 5. Then, the texture feature frames of all directions at each scale were fed into the ConvNets for training and testing after normalization process. At each scale, a frame of texture image is obtained by all directions of the texture image, which are summed and normalized. The formula is expressed as

$$F_i = \frac{\sum_{m=1}^{M} f_i^m}{M} \tag{15}$$

Where M is representing the number of filter orientations and set to 36, and **f** refers to the $\mathbf{m^{th}}$ texture feature image at $\mathbf{i_{th}}$ scale; $\mathbf{F_i}$ is the texture feature image at the i^{th} scale. Through the five-scale Gabor filter bank, five- texture feature image is acquired after input image is filtered and five prediction results are obtained after the five- texture feature image is sent to the VP detector. The final VP location is represented by the mean value of the five prediction results, as shown in the following

$$VP = \left(p_x, p_y\right) = \left(\frac{\sum_{i=1}^{N} p_i}{N} * I_w, \frac{\sum_{j=1}^{N} p_i}{N} * I_H\right) \tag{16}$$

Where N=5 scales, $\mathbf{I_W}$ and $\mathbf{I_H}$ are the width and height of the input image, respectively, and (**px, py**) is the **VP** location. Fig. 6 is illustrated different road types and the results of the estimated vanishing point by using the ConvNets algorithm.



.

Figure 6. The examples of vanishing point estimation

D. Estimate Road Boarders

In this paper, a similar technique is used as proposed in [7] to find the two most dominant borders from a set of n=35 imaginary rays that emanating from the estimated VP. The difference is that we just roughly estimate the dominant borders in order to define the sample region rather than to segment the traversable region. Let $\mathbf{N_{i,R}}$ and $\mathbf{N_{j,L}}$ are the two direct neighboring regions on either side of the ray. Fig. 7 is shown two most dominant borders and the imaginary rays. The right rays R_i (i=2, 3, 4, ..., 17) and left rays R_j (j=18, 19, 20, …, 35) with yellow rays starting from the vanishing point. In addition, the last imaginary ray (R_1) is considered to be the main bisector of the two boarders. The density of the two regions is given by

$$density\left(N_{j,L} + N_{i,R}\right) = \frac{\mathrm{O}\left(N_{j,L} + N_{i,R}\right)}{\mathrm{H}\left(N_{j,L} + N_{i,R}\right)} \tag{17}$$

Where **O** is the number of orientation consistent pixels in regions and **H** is the total number of pixels in this region. The color difference between two regions for each channel of color space is defined as,

$$\Delta f\left(NR_{j,L}, NR_{i,R}\right)_{channel} = \frac{\left|\mu(NR_{j,L}) - \mu(NR_{i,R})\right|}{\sqrt{\sigma^2(NR_{j,L}) - \sigma^2(NR_{i,R})}} \tag{18}$$

Then, the right boarder (**RB)** and left border (**LB)** are found with simply defined as the following expressions,

$$\Delta f\left(N_{RB,L}, N_{RB,R}\right) = \max\left\{\Delta(N_{i,L}, N_{i,R})_{Channel}\right\}_{Channel \in RGB} \Big|_{i=2,,17} \tag{19}$$

$$\Delta f\left(N_{LB,L}, N_{LB,R}\right) = \max\left\{\Delta(N_{i,L}, N_{i,R})_{Channel}\right\}_{Channel \in RGB} \Big|_{i=18,.35} \tag{20}$$

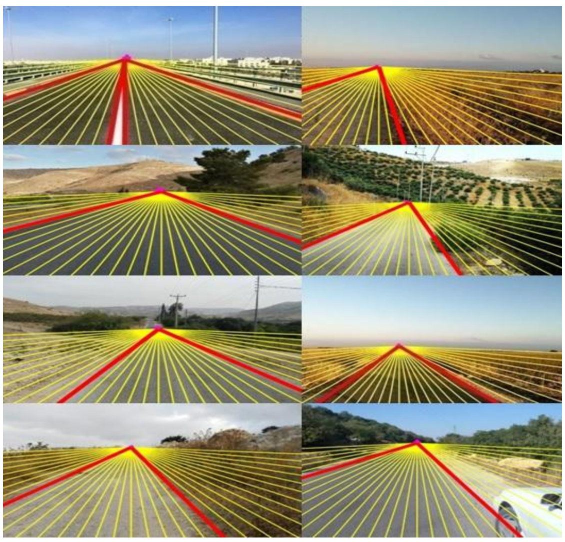


Figure 7. The imaginary rays (yellow) and boarders (red).

E. Selection of the Sample Region

In unstructured scenes, the road's appearance (e.g., color, edge, shape) is significantly varies and affected by illumination conditions. This leads to difficulty obtaining the robust appearance model with off-line training.

Therefore, it is more plausible to construct the model directly and adaptively from the input image. In this paper, the estimated VP is used to adaptively define sample region because it is able to provide a strong clue about the true location of road area. Even though the road may have various shapes, its main part can be approximated with straight borders. Thus, it is possible to represent the border of the sample region using imaginary rays. The two borders are roughly estimated to define the sample region. Thus, it has been considered 35 evenly distributed imaginary rays (yellow), as is shown in Figure 8a. The selected seed point p_{seed} is assumed at 2/3 from the VP and located at the bisector ray, and then a sample region of 10x10, which is surrounding p_{seed}, is selected as it is illustrated in Figure 8b. For all pixels in the sample region, the initial mean and initial variance are given as the following

$$\mu_o = \frac{1}{n_p} \sum_{i=1}^{n} d_{p,i} \tag{21}$$

$$\sigma_o = \sqrt{\frac{1}{n_p} \sum_{i=1}^{n_p} \left(d_{p,i} - \mu_o\right)^2} \tag{22}$$

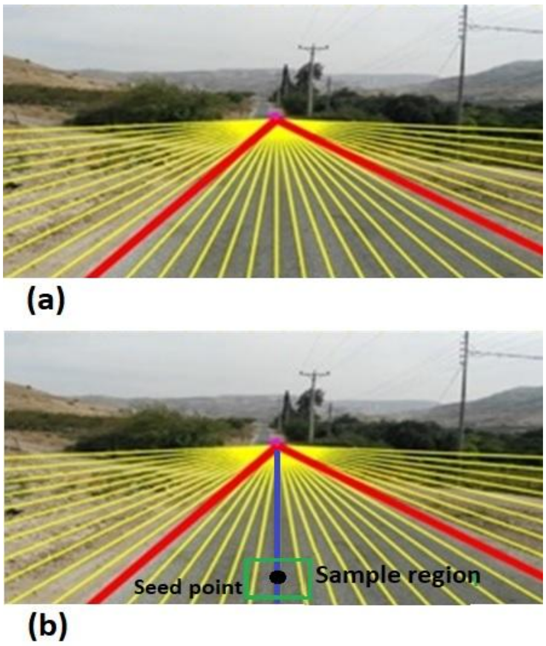


Figure 8. The selected sample region to construct appearance model.

F. Road Segmentation

In this paper, the model is based on multivariate Gaussian and considers two color spaces as the complementary features: the red–green–blue (RGB) and the illumination invariant space (IIS). In illumination conditions and shading, IIS is less sensitive than RGB. The conversion from the RGB is given as the following

$$C = \begin{cases} C_1 = \arctan\{R/\max\{G,B\}\} \\ C_2 = \arctan\{G/\max\{R,B\}\} \\ C_3 = \arctan\{B/\max\{R,G\}\} \end{cases} \tag{23}$$

Therefore, given this sample region, the mean feature vector μ_C and the covariance matrix C are first obtained using RGB channels of each pixel

$$\mu_C = \frac{1}{n_P} \sum_{i=1}^{i=n_P} Px_{C,i} \tag{24}$$

$$\Sigma C = \frac{1}{n_P} \sum_{i=1}^{i=n_P} \left[Px_{C,i} Px_{C,i}^T - \mu_C \mu_C^T\right] \tag{25}$$

$$C = \{R, G, B, C_1, C_2, C_3\} \tag{26}$$

Where n_p is pixels number in the sample region and $Px_{C,i}$ is i^{th} value of pixel for channel C.

The segmentation process starts from the seed pixel with the region growing method (8-connected neighborhood), as shown in Fig. 9. The input image is divided into five parts by the borders, the horizontal line and the vanishing point. Furthermore, in order to obtain more accurate and robust to noise results in the segmentation stage, adaptive threshold has to be applied. This threshold changes with pixel location because the pixel likelihood that belongs to the road region varies for different areas of the image. In order to compute this threshold value trapezoidal fuzzy membership function has been designed based on the possibilities of pixel location. There are three possibilities for each pixel in the image as it is illustrated in Figure 9. First, it could be located in part 1 or part 4 as a non road region. The possibility weight to consider as road region increases as **Θ_1** increases and it decreases while **Θ_4** increases. Second, when the pixel is located in part 2 or part 3 in the range [**Θ_2, $\Theta_2 + \Theta_3$**], it considers as road region with 100% certainty. Third, the pixel which belongs to part 5 is considered as a non road region with 100% certainty. Regarding to these possibilities cases analysis, it is easy to represent the model as summarized Table I. Then, the trapezoidal possibilities histograms are designed depend on the possibilities location and orientation of the pixel as well as the given regions.

TABLE I. THE POSSIBILITIES DISTRIBUTION

	Possibilities (π_ψ)		
	0	**1**	**]0, 1[**
1	$(\theta_2, +\infty)$	$(-\infty, 0)$U$(0, \theta_1)$	(θ_1, θ_2)
2	$(-\infty, \theta_1)$U$(\theta_4, +\infty)$	(θ_2, θ_3)	(θ_1, θ_2)U(θ_3, θ_4)
3	$(-\infty, \theta_3)$	$(\theta_4, +\infty)$	(θ_3, θ_4)

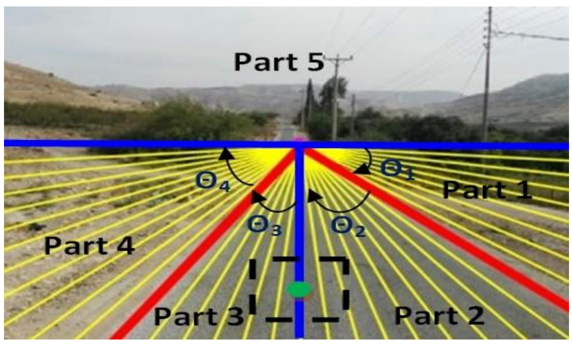


Figure 9. The possibilities of region segmentation.

As it is shown in Fig. 10 and Fig. 11, the possibilities histograms given in Table I are easily converted to trapezoidal fuzzy membership functions. This transformation can be done without any changes since both of them have same mathematical descriptions [20, 21].

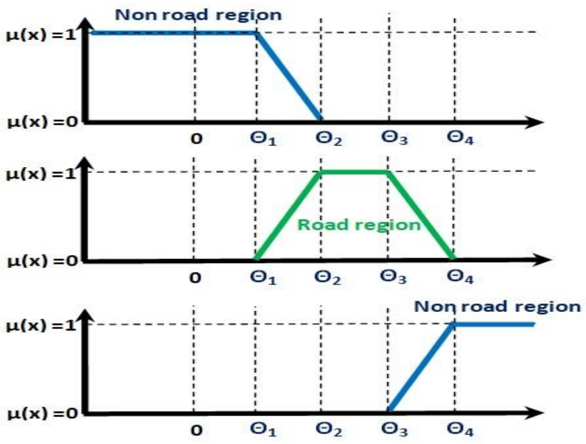


Figure 10. Trapezoidal fuzzy membership functions for 3 cases.

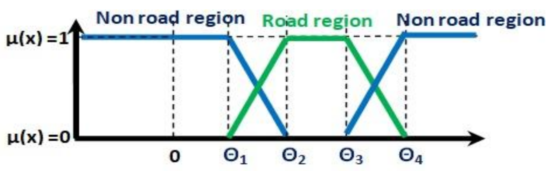


Figure 11. Total trapezoidal fuzzy membership functions

Because the possibility of each road region pixel is changed for different areas, the threshold (δ) will be varied. Fig. 12 is shown the threshold fuzzy membership functions which is experimentally modeled by using the frequency distributions [22]. The red line represents the road region whilst the blue on is for the non road region. Finaly, in order to estimate the threshold value the T-norms should be used as it is shown in Fig. 13 and it is given as the follows

$$\mu_{\delta}(x)=\sup\{\mu(\delta,\Theta)\}=\sum_{\mu=1}^{4}\min(\delta) \tag{27}$$

$$\Pi_{\delta}=\bigcap_{\sup}\Pi\delta \tag{28}$$

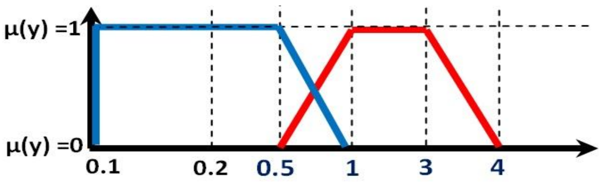


Figure 12. Threshold fuzzy membership functions

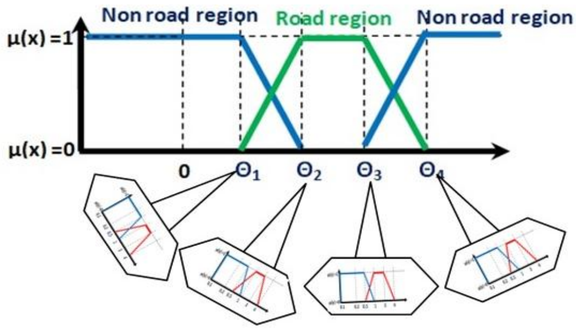


Figure 13. The T-norms process.

For example, suppose that the estimated orientation (ψ) is between θ_2 and θ_3 with membership function (green color)= {**Θ1**, **Θ2**, **Θ3**, **Θ4**} and the values of μ(x) = {0, 1, 1, 0} with threshold values μ(y) = {0.5, 1, 3, 4}, then, the threshold value can be summarized as Table II.

TABLE II. POSSIBILITIES FOR THRESHOLD

	Θ1 μ(x)=0	Θ2 μ(x)=1	Θ3 μ(x)=1	Θ4 μ(x)=0
Threshold, μ(y)=0	0.5	0.5	0.5	0.5
Threshold, μ(y)=1	1	1	1	1
Threshold, μ(y)=1	3	3	3	3
Threshold, μ(y)=0	4	4	4	4

Then, using T-norms the estimated value of the threshold can be computed as:

$$\delta=\frac{1\times(1+1+3+3)+0.5\times(17)}{1\times4+0.5\times8}=2.0625$$

Furthermore, the mahalanobis distance of P_{test} has to be computed and the estimated orientation (ψ) of the P_{test} is found by considering the color tensor **T**.

$$d_{P_{test}}=\sqrt{(P_{test}-\mu_C)^T\sum_C^{-1}(P_{test}-\mu_C)} \tag{29}$$

$$\begin{pmatrix}T_{xx} & T_{xy}\\ T_{yx} & T_{yy}\end{pmatrix} where \begin{cases} T_{xx}=V*\left[\sum_{k=1}^{3}D_{k,x}\circ D_{k,x}\right]\\ T_{yy}=V*\left[\sum_{k=1}^{3}D_{k,y}\circ D_{k,y}\right]\\ T_{xy}=V*\left[\sum_{k=1}^{3}D_{k,x}\circ D_{k,y}\right]\end{cases} \tag{30}$$

$$\psi=\frac{1}{2}\arctan\left(\frac{2T_{xy}}{T_{xx}-T_{yy}}\right)+\frac{\pi}{2} \tag{31}$$

Where $D_{k,y}$ and $D_{k,x}$ denote the derivatives of color channel along the vertical and the horizontal directions, respectively, V is convolution kernel of a smoothing filter.

The pixel P_{test} will classified into the road region if it satisfies the condition in (32).

$$if\,|d_{P_{test}}-\mu_{d,k}|<\delta\sigma_{d,k} \tag{32}$$

Where δ is the threshold that depends on location of pixel and it ensures the accuracy of the segmentation against the noise, $\mu_{d,k}$ is the mean of Mahalanobis distance and $\sigma_{d,k}$ is the variance of Mahalanobis.

Moreover, once the new pixel represents a road region, the variance and mean has to be updated and the pixel will add to road region.

$$\mu_K = \frac{\left[\mu_{K-1} N_{K-1} + d_{P_{test}}\right]}{N_{K-1}+1} \tag{33}$$

$$\sigma_K = \sqrt{\frac{\sigma_{K-1}^2 N_{K-1} + \left(d_{P_{test}} - \mu_K\right)^2}{N_{K-1}+1}} \tag{34}$$

Where N_{K-1} is the total number of pixels in current road region.

As it is shown in Fig.14, the adaptive trapezoidal fuzzy model and multivariate Gaussian model combining features are successfully and robustly segmented the road. However, the training region may contain non-road pixels and could not always represent the exact road region. For instance, the segmentation for the road on (a, b, and c) in Fig. 14 is slightly not satisfying because the sample region included a small portion of non-road pixels because the road is classified as curved road and this type of road tends to compute inaccurate VP and dominant borders. Moreover, as can be seen in Fig. 15, even though the estimated VP and the dominant border detection were not good enough, the road region can still be correctly found with slightly inaccurate result and the obtained results of segmentation is not be highly influenced by this condition.

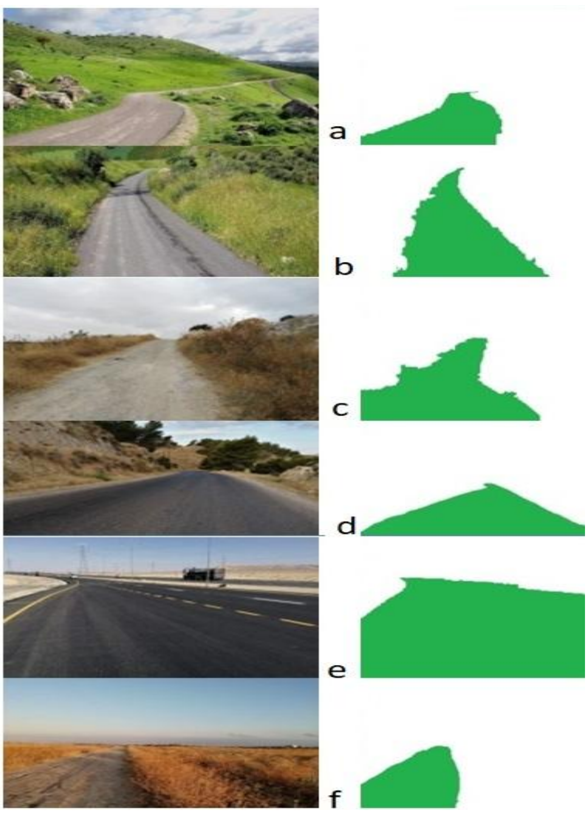


Figure 14. Example of segmentation.

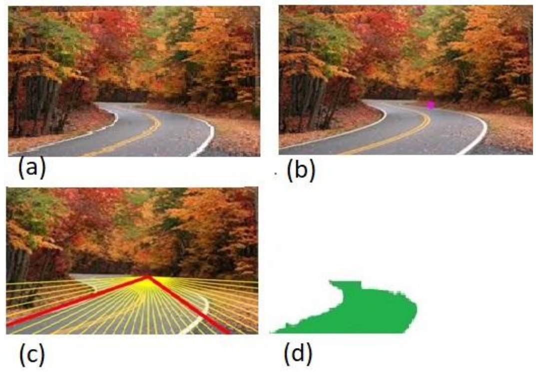


Figure 15. Example of segmentation under condition of inaccurate VP and dominant borders. a : original image, b: inaccurate VP, c inaccurate dominant boarders, d: final segmentation

G. Challenged Roads by Fuzzy C-mean

In real life, the road may have more curved challenged borders as shown in Fig. 16, then, two boarders road estimator is not 100% sufficient. Hence, the segmentation result is slightly affected by this condition as it is described in previous section.

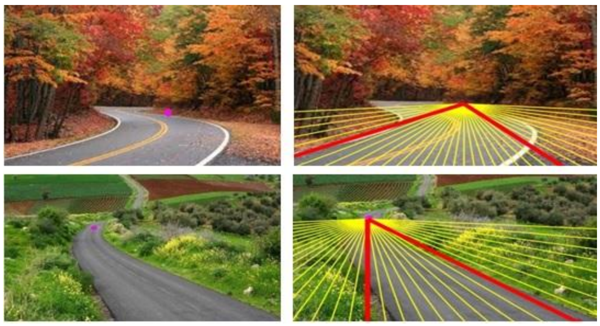


Figure 16. The examples of more challenged curved roads.

In addition, the segmentation based on the pixel grids or based upon the local spatial information always leads to high computational complexity. This is due to the iteration process to compute the distance between pixels within spatial neighbors and clustering centers. The superpixels using SLIC method tends to preserve boundaries and they have color and texture uniform that leads them to be more meaningful and efficient [23, 24]. However, the improved Fuzzy C-mean (FCM) algorithm based upon morphological reconstruction and membership filtering is significantly faster and more robust [25, 26]. If the local spatial information of an image is computed before using FCM, the computational complexity would be efficiently reduced especially for more challenged roads. In addition, modification the membership function without modifying the objective function will leads to get simple and faster algorithm [25]. Motivated by [26, 27], in this research, the improved FCM algorithm has been implemented. It is fast superpixel algorithm which able to provide better pre-segmentation results with shorter execution time. The objective function for color image segmentation given as the following:

$$J_m = \sum_{L=1}^{N}\sum_{k=1}^{c} P_L u_{kL}^m \left\| \left(\frac{1}{P_L} \sum_{p\in R_L} x_p \right) - v_k \right\|^2 \quad (35)$$

Where **L** is the color level, $1 \le L \le N$, N is the number of regions of the superpixel image, $\mathbf{P_L}$ is the number of pixels in the $\mathbf{L^{th}}$ region $\mathbf{R_L}$, $\mathbf{x_p}$ is the color pixel within the $\mathbf{L^{th}}$ region of the superpixel image, and **m** is the weighting exponent. The unconstrained optimization problem can be obtained by utilizing the Lagrange multiplier technique which able to minimizes the utilizing objective function

$$\breve{J}_m = \sum_{L=1}^{N}\sum_{k=1}^{c} P_L u_{kL}^m \left\| \left(\frac{1}{P_L} \sum_{p\in R_L} x_p \right) - v_k \right\|^2 - \omega\left(\sum_{k=1}^{c} u_{kL} - 1 \right) \quad (36)$$

Where ω is Lagrange multiplier. The corresponding solutions for $\mathbf{u_{kL}}$ and $\mathbf{v_k}$ are achieved by computing the differential equation $\mathbf{J_m}$:

$$v_k = \frac{\sum_{L=1}^{N} u_{kL}^m \sum_{p\in R_L} x_p}{\sum_{L=1}^{N} P_L u_{kL}^m} \quad (37)$$

$$u_{kL} = \frac{\left\| \left(\frac{1}{P_L} \sum_{p\in R_L} x_p \right) - v_k \right\|^{-2/(m-1)}}{\sum_{j=1}^{c} \left\| \left(\frac{1}{P_L} \sum_{p\in R_L} x_p \right) - v_k \right\|^{-2/(m-1)}} \quad (38)$$

Where $\mathbf{u_{kL}}$ represents the fuzzy membership of gray value ***L*** with respect to the $\mathbf{k_{th}}$ clustering center $\mathbf{v_k}$.

It is worth to mention here that after the implementation of the FCM algorithm, the results will be segmented by using the proposed method mention earlier. Therefore, Fig. 17 is illustrated the challenging problem for road edge detection using improved Fuzzy C-Means when the road in input image is classified as challenged curved road. In addition, Fig. 18 is shown comparison examples of segmentation between (improved FCM combined with the trapezoidal fuzzy membership function segmentation method) and (the VP with two dominants boarders combined with trapezoidal fuzzy membership function segmentation method). It is clearly shown that the improved FCM combined with trapezoidal fuzzy membership functions is slightly given better results in more challenging curved road.

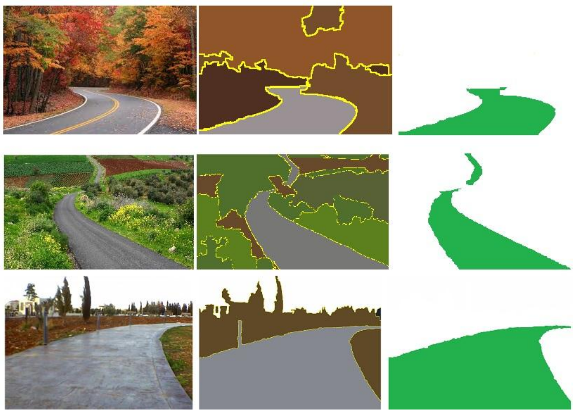


Figure 17. Examples of segmentation by improved FCM

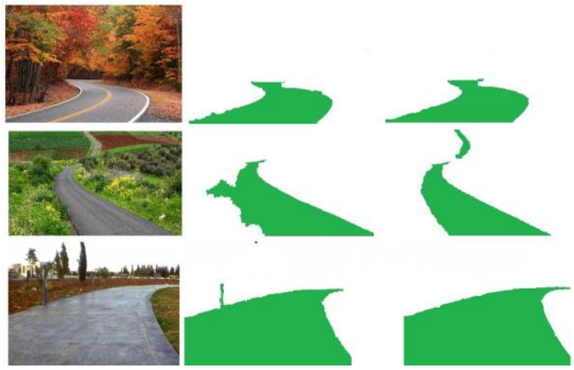


Figure 18. Example of segmentaion. Column 1: input image, Column 2: segmentation by VP and two dominant method, Column 3: segmentation results by improved FCM.

III. Experimental Results and Discussion

Three experiments have been carried out in order to evaluate the proposed method. Firstly, VP detection had been tested on the dataset 1. This dataset consists of 4000 images of unstructured environment under various conditions. Among this dataset a total of 560 images were taken in Eastern Jordan desert. Then, to evaluate and demonstrate the performance of the traversable region segmentation another challenging image dataset 2 was also used for more intensive tests on different unstructured scenarios. This dataset contains 5000 images taken along a Northern Jordan (Irbid). Moreover, all images were normalized to same size (240x180) and all the algorithms were run on a personal laptop (Intel i5-3570 CPU) without GPU acceleration. Lastly, for the purpose of showing the effectiveness of the method for real-time application, the proposed framework was implemented on a Bogie rover robot platform in an unstructured King Abdullah II Gardens environment.

A. *Image Dataset*

We have created two challenging datasets on different unstructured scenarios. The first dataset consists of 4000 images and the second one consists of 5000 images. They are available at *https://www.kaggle.com/ramiahmad/two-datasets-for-visionbased-realtime-detection*. They were taken under various environmental conditions and various road classifications such as different day times, indoor/outdoor scenes, shapes (normal or curved), and various surface structures (soil, asphalt). In order to enable quantitative performance evaluation, we manually annotated the vanishing point in each image. The Statistics regarding the road surfaces and the lighting conditions are given in Table III.

TABLE III. Statistics of the Datasets

Description	Number of images		Percentage (%)	
	Dataset1	Dataset2	Dataset1	Dataset2
Brick surfaces	625	872	15.6	17.44
pavement surfaces	712	953	17.8	19.06
Indoor surfaces	152	245	3.8	4.9
Soil surfaces	727	1089	18.18	21.78
Asphalt surfaces	1804	1841	45.1	36.82

© 2021 Int. J. Mech. Eng. Rob. Res

Normal lighting	3025	4121	75.63	82.42
Extreme lighting	975	879	24.37	17.58
Non curved roads	1525	1827	38.125	36.54
Curved roads	2475	3173	61.875	63.46

B. Performance of the Proposed Detector

In order to evaluate the machine learning models on any dataset, the K-folds cross validation has to be used to verify the algorithm. Randomly, the dataset is partitioned into K=5 subsets. The training data are K-1 (4 sub-samples), the testing data is the remaining part to validate the performance of the system. Then, the folds are repeated K times and the K results will be averaged to estimate the final result. The final result was represented by the average value of the prediction results. In order to quantitatively evaluate the vanishing point, the distance between the predicted VP location and ground-truth location was computed and the estimation error is given as [28].

$$D = |VP_{\mathrm{detected}} - VP_{groundtruth}| = \begin{cases} true.VP & D < Th \\ false.VP & D > Th \end{cases} \tag{39}$$

$$E_{vp} = \frac{|VP_{\mathrm{detected}} - VP_{groundtruth}|}{L_d} \tag{40}$$

Where VP_{detected} and VP_{ground} truth are the detected vanishing point and ground-truth respectively, L_d is the image diagonal length. The vanishing point detected value is considered as true vanishing point if the distance is smaller than a threshold value. Thus, by changing the threshold value from 0% to 42% of the image width, the accuracy rate of the tested samples was obtained as shown in Fig. 17 and Fig. 18. In order to verify the performance of the proposed ConvNets detector, several tests for the algorithm on different resolutions have been carried out as shown in Fig. 19 and Fig. 20. The impact from the kernel size of the Gabor filter has been considered with two kernel sizes (5 and 10). According to the performance result curves, it was observed that the ConvNets detector exhibited similar performance when the resolution was reduced from 100×100 to 30×30 but poor performance when the resolution was 10×10. The metric degree of error (DoE) was introduced to intuitively compare the effects of different kernel sizes and resolutions on the VP detector. With decreasing in the DoE, the VP detector achieved 100% accuracy.

Thus, the larger the resolution is, the better the detector performs. However, the performance was a little better under the kernel size of 5 than size 10 because the bigger size tends to blur the texture of the image. An experiment was carried out in order to compare the performance of the proposed vanishing point algorithm with other algorithms such as Gaussian Sphere [29], Interior and Exterior Region [30], Generalizing Laplacian of Gaussian Filters [31], and Expectation Maximization [32]. The Gaussian sphere takes the Gaussian as accumulated space. The Interior and Exterior Region algorithm realizes the VP detector using one-dimensional (1D) histogram to detect the three orthogonal VP from the both interior and exterior regions. The VP could be detected based on the estimated texture by the Generalizing Laplacian of Gaussian Filters which applies a filter to estimate the texture orientation at each pixel of an image. In addition, the Expectation Maximization algorithm is an algorithm that uses an efficient approach to detect the VP from a single view assuming an un-calibrated camera in the environments. The combative accuracy between the proposed method and other methods is illustrated in Fig. 21. Even though the Gaussian Sphere is out performed than the other three methods, but its accuracy is still less than 100% even if the threshold was relaxed to 40%. This means that these three methods have relatively poor performance if the test is setting with serious barrel distortion. Furthermore, the accuracy result obtained from the proposed method is better than others over the whole range of the image width ratio.

Moreover, some performances examples for vanishing point estimation with different methods are shown in Table IV. This table based on two different datasets with different methods in terms of accuracy and runtime. It is clear that the average error of the proposed method is less than others. Thus, this method has higher performance with respect to accuracy. In addition, the computation time of the proposed method was significantly the shortest one. Therefore, the proposed method has able to achieve the best tradeoff between accuracy and time efficiency.

Table IV shows the performance of different VP algorithms on the test set of 1000 images for each Dataset. The average errors of the proposed method (0.0869 and 0.0965) were significantly lower than that of the MikSik (0.1211 and 0.1752) and the Alvarez (0.1154 and 0.1862). The Alvarez algorithm combines the illuminant-invariant feature space with a road class-likelihood classifier in a frame-by-frame framework to provide reliable road detection results despite lighting variations and shadows. The Miskis is designed for speed up the voting scheme and implement the expansion Gabor wavelets into a linear combination of Haar like functions to fast filtering the integral image. Furthermore, the proposed method also has lower average error compared to Gabor- based method (0.1039 and 0.1778). Moreover, the Gabor-based method calculates voting score for each VP candidate from all pixels in half-disk region, and therefore it is affected by clutter pixels. In addition, the Gabor-based method uses only intensity for computing edge orientations and magnitudes. In comparison, the proposed method uses only ConvNets predictor for estimating VP, and therefore significantly it reduces the computation load. It applies the nonlinear Rectified Linear Units to increase the nonlinear properties of the model and provide ability to train much faster without making a significant difference to the accuracy. Moreover, the proposed method employs two color spaces as the complementary features to find the edge pixels and their orientations (via color tensor). Therefore, it could distinguish color pixels even if they have similar intensity. For images of size 100 ×100 pixel, the average processing time per image of proposed method (0.0121

© 2021 Int. J. Mech. Eng. Rob. Res

sec) was significantly shorter than that of the Gabor-based method (7.315 sec), Alvarez (3.1221 sec), and Miskis (2.4217 sec). As a result, the proposed method has higher performance with respect to accuracy. In addition, the computation time of the proposed method was significantly the shortest one. Therefore, the proposed method has able to achieve the best tradeoff between accuracy and time efficiency.

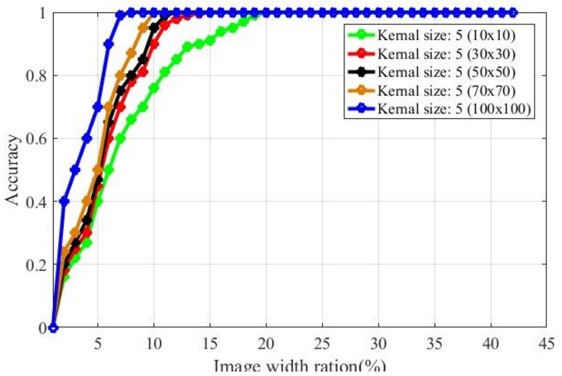


Figure 19. Prediction accuracy on multi-resolution with kernel size 5

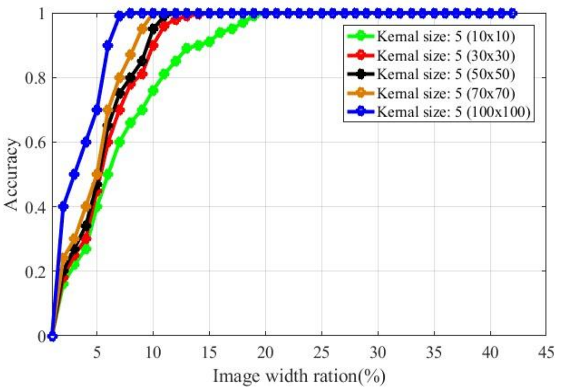


Figure 20. Prediction accuracy on multi-resolution with kernel size 10

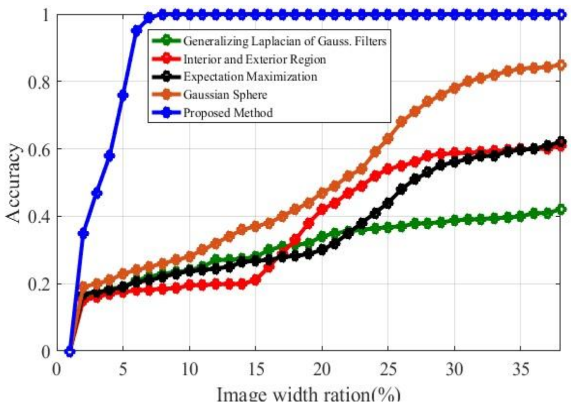


Figure 21. The prediction accuracy comparison

TABLE IV. COMPARISON PERFORMANCE OF THE VANISHING POINT.

Method	Average Error		Computation Time (sec)
	Dataset1	Dataset2	
Kong[7]	0.1039	0.1778	7.315
Alvarez[12]	0.1154	0.1862	3.1221
MikSik [9]	0.1211	0.1752	2.4217
Proposed method	0.0869	0.0965	0.0121

Moreover, it has been selected the five most often used quantitative parameters for algorithms evaluation [33]. Given a classifier and a set of instances (the test set) for the proposed road detector, there are four possible outcomes. When the instance is positive and has been classified as positive, it is considered as true positive (TP) that is described the number of road pixel correctly identified. Then, when it is classified as negative, it is considered as a false negative (FN) that is the number of road pixels incorrectly recognized. If both the instance and it is classified are negative, it is considered as a true negative (TN) that expresses the number of background pixels correctly detected. If the instance being a false and it is classified as positive, it will be considered as a false positive (FP) that describe the number of background pixels incorrectly detected. The contingency table which is constructed to represent the dispositions of the set of instances and to illustrate the evaluations of the proposed algorithm is shown in Table V. The results of quantitative contingency are shown in Table VI. The proposed method has a sensitivity of 92.7%, a specificity of 94.9%, F-measure of 91.7%, precision of 91.2, and accuracy of 92.4%. The Accuracy value of 92.40% of the proposed algorithm means that only 7.6% of the images have been classified incorrectly to the other class and this over-classification is not critical. The Sensitivity value of 92.7% means that most actual images have been classified correctly. Also, the high value of the F1-measure parameter provides a crucial sight to overall performance of the proposed algorithms. The proposed method outperformed the Miskis method which has a sensitivity of 85.4%, a specificity of 88.4%, F-measure of 85.9%, precision of 87.4, and accuracy of 86.7%. Furthermore, the proposed method outperformed the Kong method which has a sensitivity of 76.5%, a specificity of 69.7%, F-measure of 75.6%, precision of 74.1, and accuracy of 66.7%. The Kong method uses only the color and orientation properties of road borders, so it is susceptible to background edges. In contrast, the proposed method employs the properties of not only road borders but also road regions. The proposed method also had better results than the Alvarez method. In [7, 9], the two algorithms may work badly in curved road boundaries and the method in [12] may fail if the road and background are very similar. As a result, it is clearly obvious that the overall performance of our proposed algorithm outperforms the others. And it can exhibit high robustness against the variations of the road scene.

TABLE V. THE CONTINGENCY TABLE

Parameter	Symbol	Equation
Sensitivity	S	$\frac{TP}{TP+FN}$
Specificity	SP	$\frac{TN}{TN+FP}$
False alarm	F	$1-specificity$
Precision	P	$\frac{TP}{TP+FP}$
Accuracy	A	$\frac{TP+TN}{TP+TN+FP+FN}$
Successively	*F1-measure*	$\frac{2.TP}{2.TP+FP+FN}$

TABLE VI. THE CONTINGENCY RESULTS

Method	S (%)	SP (%)	F1 (%)	P (%)	A (%)
Kong [7]	76.5	69.7	75.6	74.1	66.7
Alvarez [12]	86.4	85.7	84.2	82.7	85.4
MikSik [9]	85.4	88.4	85.9	87.4	86.7
proposed method	92.7	93.2	91.7	91.2	92.4

Recently, the large majority of state of- the-art algorithms for road detection use machine learning techniques. Thus, five different networks methods were considered and were also evaluated on challenging Dataset 2 benchmark. The significant Up-conv-Poly is proposed as convolutional network for road segmentation and it increased the width of the up-convolutional side of the network in order to improve the model accuracy [34]. Then, DEEP-DIG [35] used a full convolutional architecture and multiple upscaling steps for image interpolation. It uses geometric transformations (perspective transformation and pixel changes) in order to obtain better road segmentation results. Recently, fusing data point clouds and camera sensor images for road detection has been proposed in [36]. In this LidCamNet method, the sparse point clouds are projected onto image plane and up-sampled to obtain a set of dense 2D images, encoding spatial information. Then, it is trained to carry out road detection by using the fusing data. Chen et al. in [37] developed a deep neural network within a Bayesian framework to jointly estimate the road surface and its boundaries. Also, the fully convolutional encoder–decoder that contains an intermediate context module was proposed in [38] in which the LIDAR point clouds are transformed into 2D top-view images that are then used as input for an FCN to carry out road segmentation. The performance of the proposed method on the road category is reported in Table VII together with the results obtained by other state-of-the-art approaches. As can be seen, the overall best performance was achieved by the proposed method with F1-measure score of 96.14%. This is followed by the LidCamNet that obtained a F1-measure score of 95.02% and then the LoDNN at 94.34%. The worst performance was obtained by the Up-conv-Poly that only had resulting in F1-measure score of 93.27%.

TABLE VII. THE CONTINGENCY RESULTS ON DATASET2.

Method	S (%)	SP (%)	F1 (%)	P (%)	A (%)
Upconv-Poly [33]	93.41	94.25	93.27	92.45	93.65
DEEP-DIG [34]	93.51	93.78	93.42	93.67	94.76
LidCamNet [35]	94.72	95.58	95.02	94.91	94.82
RBNet [36]	**95.12**	94.72	94.07	95.11	95.14
LoDNN [37]	94.65	95.41	94.34	**96.12**	96.12
Proposed method	95.10	**96.17**	**96.14**	95.81	**96.43**

The segmentation accuracy has been quantitatively evaluated based on the matching score. This matching score will reach optimal value 1 (maximum value) when the detected areas are totally matches the ground truth values. Therefore, this score has been changed from minimum value 0 to the maximum value 1 in order to compute the segmented images rate as it is illustrated in Fig. 22. Regarding to this result, the proposed method has higher performance in comparing with other methods.

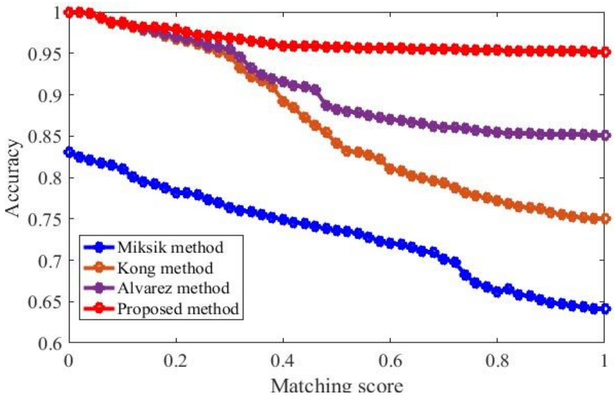


Figure 22. The accuracy of segmentation comparison.

Furthermore, for the quantitative evaluation performance, a comparison between the proposed method and the results obtained by other state-of-the-art approaches is shown in Fig. 23. The LoDNN is out performed than the other four methods and its accuracy reaches 100% when the threshold was relaxed to 12%. However, the proposed method has slightly better accuracy than in [38] and obviously is out performed the others. Thus, the accuracy result obtained from the proposed method is better than other state-of-the-art approaches over the whole range of the image width ratio.

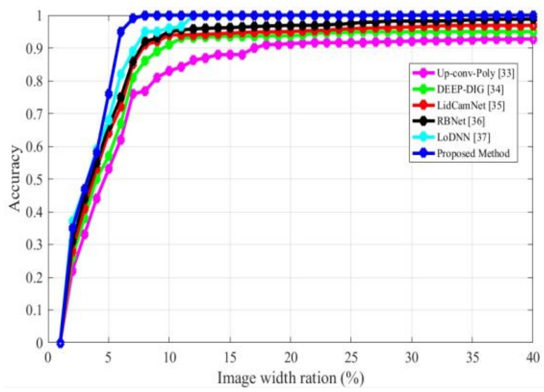


Figure 23. Quantitative evaluation performance comparison.

C. Real Time Navigation for Robot

The proposed approach has been implemented on a rover robot in the King Abdullah II Gardens environment with non-structured pedestrian lanes with 1.0–1.5 m wide. The fast stereo matching algorithm for robotic applications that is suitable for embedded real-time systems is adapted from [39]. It can achieve a high performance on the system without losing good quality of stereo matching under real world conditions. The implementations offer high flexibility in terms of image dimensions and disparity range. Also, it can successfully eliminate false positives to provide reliable 3D data. In order to reconstruct the 3D data of a scene captured from two different points of view, the pixel correspondences between both images has to be found. Once the correct disparity for a pixel p(x, y) is computed, it can be used to calculate orthogonal distance (z_c) between camera's optical center and projected scene point.

$$z_c = \frac{b.f}{d} \tag{41}$$

where d is the disparity, b the baseline and f the camera's focal length.

Then, the 3D data in camera coordinates is given as follows.

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = K^{-1} \begin{pmatrix} x.z_c \\ y.z_c \\ z_c \end{pmatrix} \tag{42}$$

where K is the camera calibration matrix, the pixel is given in homogeneous coordinates $(u\ .z_c,\ v.\ z_c,\ z_c)^T$ and z_c is calculated with (43). K and f have to be determined by camera calibration which is very important for fast stereo matching.

The rigid transformation of the center of the road target into the camera coordinate systems is homogeneously represented as

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = T^{W/C} \begin{pmatrix} x_W \\ y_W \\ z_W \end{pmatrix} \tag{43}$$

where (x_W, y_W, z_W) is the road position in the world coordinate and the transformation $T^{W/C} = T^{R/C}\, T^{W/R}$. $T^{W/R}$ is the rigid transformation from the world coordinate system to the robot coordinate. $T^{R/C}$ is the fixed transformation from the robot coordinate system to the camera coordinate system. Then, the road position, which is the traversable region 3D points p(x_w, y_w, z_w), is mapped to the world coordinate frame by (44).

$$\begin{pmatrix} x_W \\ y_W \\ z_W \\ 1 \end{pmatrix} = \left(T^{R/C} T^{W/R}\right)^{-1} \begin{pmatrix} z_c \\ y_c \\ z_c \\ 1 \end{pmatrix} \tag{44}$$

In order to estimate the pose and build the map, a feature-based Simultaneous Localization and Mapping (SLAM) system is chosen. This ORB-SLAM [40] operates in real time, in small and large environments. It is robust to severe motion clutter, and it allows wide baseline loop closing and re-localization. Due to the limited resources, e.g. memory and processing power, and to avoid producing too many 3D points of the traversable region, the voxel filter has been used to downsample the point cloud. The rover robot has precisely mapped the path so that it can correctly identify bridges and cross them. Thus, the rover robot has ability to localize itself and simultaneously build the map for its environment. The proposed approach has been implemented on a rover robot in the King Abdullah II Gardens environment with non-structured pedestrian lanes with 1.0–2 m wide. The robot was only allowed to navigate on the normal pedestrian roads, as if it was a human. The rover robot has ability to accurately localize itself and simultaneously build the traversable map for the surrounding environment. The real-time traversable region detection is shown in Fig. 24. The binocular sensor had been mounted on the rover robot platform with image frame rate 30 fps. The loop closing procedure searches and detects any loop. As it is shown in Fig. 25, there are two loops in the trajectory. The reconstruction after the loop closure shows that the whole trajectory has updated and validated. The red scattered points is shown local map, which after loop closure extends after it is closed. Thus, both the traversable region and the robot pose were corrected. Fig. 26 is shown the final 3D map without 3D scattered features points.

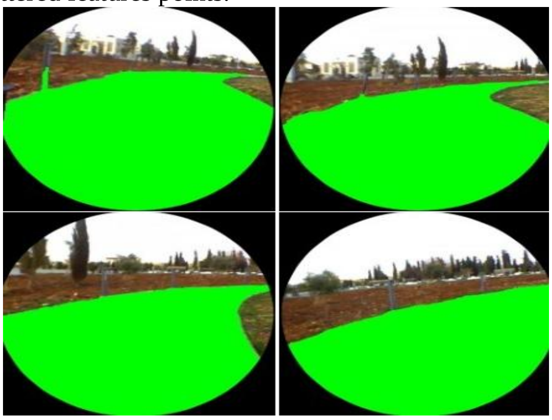


Figure 24. Real-time traversable region detection.

The results of the robot trajectory (~293m) during the experiment on 3D plot and on Google map are shown in Figs. 27 and 28, respectively. Fig. 27 is shown the built map and the rover robot poses after they are overlapped on Google map. The blue line represents the robot pose, while the green line represents the road area (pedestrian region). These results demonstrated that the rover robot is capable of navigating and generating the unstructured environment maps in real-time. It is very important to mention here that the trajectory contains a foot bridge as it is shown in Fig. 29. This foot bridge has been accurately mapped and the robot locates and crosses the bridge like a human. In the future, further researches on localization and mapping will be presented, which is now out of the scope of this research.

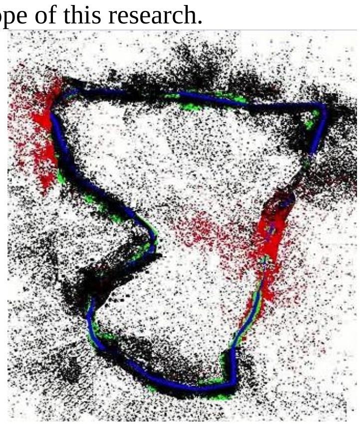


Figure 25. 3D Map after loop closure. The traversable region (green), camera pose (blue), and local map in red.

Figure 26. The final 3D map

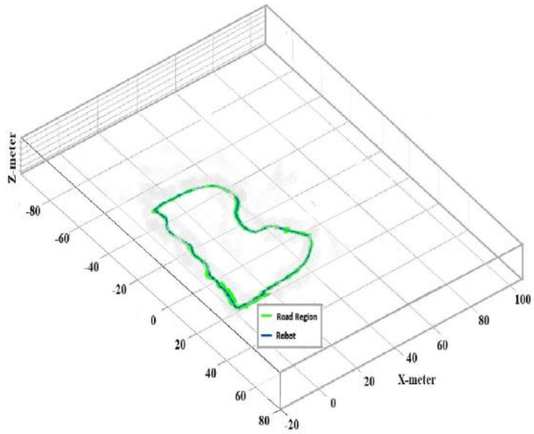


Figure 27. A complete trajectory of the rover robot in traversable region.

Figure 28. The traversable mapping overlapped on real map

Figure 29. The trajectory contains Foot Bridge.

IV. Conclusion

In this paper, we proposed a novel approach for road detection in unstructured regions for real-time navigation. In order to address the issues of a lack of clearly established road boundaries and scene variety, the issue is divided into two parts: road type classification and road area segmentation. Using the samples from the image datasets, the two road models corresponding to various types of terrain and illumination conditions were learnt. This method makes use of a novel VP detection algorithm based on ConvNets to estimate the vanishing point. Multi-resolution tests were performed to pick the required low resolution in order to meet the real-time applications without decreasing accuracy. The proposed VP detector showed good effect and capability. In addition, the evaluation indicates that the proposed model is accurate and robust. Experimental results on the public dataset have shown that the proposed method is able to detect various unstructured roads in real-time.

For normal roads, the road sample region will be determined adaptively by VP and road boundaries. To identify the road region rapidly and reliably from the sample region, a self-supervised segmentation approach was constructed based on a multivariate Gaussian model and trapezoidal fuzzy membership functions. The proposed method can detect various unstructured roads in real-time according to the experimental results. The robust and rapid fuzzy C-Means algorithm has been proposed for curved roads to improve segmentation efficiency and reduce noise impact. The results show that the algorithm performs well on curved roads and provides better segmentation results when using different images without the need to change parameters. Furthermore, the efficacy of the proposed method for robot navigation has been demonstrated on a real rover robot in difficult environments and the achieved results shows the good performance of the system. In the future, further research on localization and mapping will be presented, which is now out of the scope of this paper.

Conflict of Interest

The author declares that there are no conflicts of interest regarding the publication of this article.

Reference

[1] J. L. Mei, Y. F. Yu, H. J. Zhao, "Scene-adaptive off-road detection using a monocular camera," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 242-253, 2018.

[2] C. Rotaru, T. Graf, and J. Zhang, "Color image segmentation inh si space for automotive applications," *Journal of Real-Time Image Processing*, vol. 3, pp. 311–322, 2008.

[3] R. Christopher, "Combining laser range, color, and texture cues for autonomous road following," in *Proc. IEEE ICRA*, 2002, pp. 4320–4325.

[4] Y. He, H. Wang and B. Zhang, "Color-based road detection in urban traffic scenes," *IEEE Transactions on Intelligent Transportation Systems (TITS)*, vol. 5, pp. 309–318, 2004.

[5] S. L. Phung, M. C. Le, A. Bouzerdoum, "Pedestrian road detection in unstructured scenes for assistive navigation," *Comput. Vis. Image Underst*, vol. 149, pp. 186–196, 2016.

[6] P. D. Cristóforis, M. A. Nitsche, T. Krajník, M. Mejail, "Real-time monocular image-based path detection," *J. Real Time Image Proc*. vol. 11, pp. 335–348, 2016.

[7] H. Kong, J. Audibert, J. Ponce, "General road detection from a single image," *IEEE Trans. Image Process*, vol. 19, 2010, pp. 2211–2220.

© 2021 Int. J. Mech. Eng. Rob. Res

[8] P. Moghadam, J. A. Starzyk, W. S. Wijesoma, "Fast vanishing-point detection in unstructured environments," *IEEE Trans. Image Process*, vol. 21, pp. 425–430, 2012.
[9] O. Miksik, "Rapid vanishing point estimation for general road detection," in *Proc. the IEEE International Conference on Robotics and Automation*, St. Paul, MN, USA, 14–18, 2012.
[10] C. Tan, H. Tsai, T. Chang, M. Shneier, "Color model-based real-time learning for road following," in *Proc. the IEEE Conference on Intelligent Transportation Systems*, Toronto, ON, Canada, 17–20 September, 2006.
[11] O. Ramstrom, H. Christensen, "A method for following unmarked roads," in *Proc. the IEEE Intelligent Vehicles Symposium*, Las Vegas, NV, USA, 6–8 June 2005.
[12] J. M. Álvarez, A. M. Lopez, "Road detection based on illuminant invariance," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, 2011, pp. 184–193.
[13] X. Lu, "Self-supervised road detection from a single image," in *Proc. the IEEE International Conference on Image Processing*, Quebec City, Canada, 27–30 September 2015.
[14] O. Miksik, P. Petyovsky, L. Zalud, P. Jura, "Robust detection of shady and high-lighted roads for monocular camera based navigation of UGV," in *Proc. the IEEE International Conference on Robotics and Automation*, Shanghai, China, 9–13 May 2011.
[15] J. M. Álvarez, A. M. Lopez, T. Gevers, F. Lumbreras, "Combining priors, appearance, and context for road detection," *IEEE Trans. Intell. Transp. Syst.* vol. 15, 2014, pp. 1168–1178.
[16] L. Xiao, B. Dai, D. X. Liu, D. W. Zhao, T. W, "Monocular road detection using structured random forest," *Int. J. Adv. Robot. Syst.* vol. 13, pp. 1–13, 2016.
[17] G. L. Oliveira, W. Burgard, T. Brox, "Efficient deep models for monocular road segmentation," in *Proc. the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 9–14 October 2016.
[18] V. Mnih, K. Kavukcuoglu, D. Silver, et al. "Human-level control through deep reinforcement learning," *Nature*, vol. 518, 2015, pp. 529–533.
[19] J. Li, X. Mei, and D. Prokhorov, "Deep neural network for structural prediction and lane detection in traffic scene," *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 99, pp. 1–14, 2016.
[20] R. Al-Jarrah, H. Roth, "Design blimp robot based on embedded system and software architecture with high level communication and fuzzy logic," in *Proc. the 9th International Symposium on Mechatronics and its Applications (ISMA)*, 2013, pp. 1-6.
[21] R. Al-Jarrah, R. A. Jellal, H. Roth, "Blimp based on embedded computer vision and fuzzy control for following ground vehicles," *IFAC Proceedings Volumes*, vol. 46, no 29, pp. 7-12, 2013.
[22] R. Al-Jarrah, H. Roth, "Developed blimp robot based on ultrasonic sensors using possibilities distribution and fuzzy logic," *Journal of Automation and Control Engineering*, vol. 1, no. 2, pp. 119-125, 2013
[23] C. Guo, S. Mita, and D. A. McAllester, "Adaptive non-planar road detection and tracking in challenging environments using segmentationbased markov random field," in *Proc. IEEE ICRA*, 2011, pp. 1172–1179.
[24] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. S¨usstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, pp. 2274–2282, 2012.
[25] T. Lei, P. Liu, X. Jia, X. Zhang, H. Meng, and A. K. Nandi, "Automatic fuzzy clustering framework for image segmentation," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 9, pp. 2078-2092, Sept. 2020.
[26] T. Lei, X. H. Jia, T. L. Liu, S. G. Liu, H. Y. Meng, and A. K. Nandi, "Adaptive morphological reconstruction for seeded image segmentation," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5510-5523, Nov. 2019."
[27] T. Lei, X. H. Jia, Y. N. Zhang, S. G. Liu, H. Y. Meng, and Asoke K. Nandi, "Superpixel-based fast fuzzy c-means clustering for color image segmentation," *EEE Transactions on Fuzzy Systems*, vol. 27, no. 9, September 2019
[28] P. Moghadam, J. A. Starzyk, W. S. Wijesoma, "Fast vanishing-point detection in unstructured environments," *IEEE Trans. Image Process*, vol. 21, pp. 425–430, 2012.
[29] S. T. Barnard. Interpreting perspective images, Artif Intell1983; 4: 435–462.
[30] B. Li, K. Peng, X. Ying, et al. "Vanishing point detection using cascaded 1D hough transform from single images," *Pattern Recogn Lett*, 2012, vol. 33, pp. 1–8.
[31] H. Kong, S. E. Sarma, and F. Tang, "Generalizing Laplacian of Gaussian filters for vanishing-point detection," *IEEE T Intell Transp*; vol. 14, pp. 408–418, 2013.
[32] J. Kosecka and W. Zhang, "Video compass," in *Proc. the 7th European conference on computer vision*, vol. 4, Copenhagen, Denmark, 28–31 May 2002, pp.476–490. Berlin, Heidelberg: Springer-Verlag.
[33] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.
[34] G. L. Oliveira, W. T. Burgard, T. Brox., "Efficient deep models for monocular road segmentation," in *Proc. the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 9–14 October 2016, pp. 4885–4891.
[35] J. Munozbulnes, C. I. Fernandez, I. Parra, D. Fernandezllorca, M. A. Sotelo, "Deep fully convolutional network with random data augmentation for enhanced generalization in road detection," *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, Japan, pp. 366–371, 2017
[36] L. Caltagirone, M. Bellone, L. Svensson, M. J. R. Wahde, "Systems, a. LIDAR-camera fusion for road detection using fully convolutional neural network," *Robot. Auton. Syst.*, 2019, vol. 111, pp. 125–131.
[37] Z. Chen, Z. Chen, RBNet: A deep neural network for unified road and road boundary detection, in: International Conference on Neural Information Processing, Springer, 2017, pp. 677–687.
[38] L. Caltagirone, S. Scheidegger, L. Svensson, M. Wahde, "Fast lidar-based road detection using fully convolutional neural networks," in: Intelligent Vehicles Symposium (IV), 2017 IEEE, IEEE, 2017, pp. 1019–1024.
[39] M, Humenberger, C. Zinner, M. Weber, W. Kubinger, M. Vincze, "A fast stereo matching algorithm suitable for embedded real-time systems," *Comput. Vis. Image Underst*. vol. 114, pp. 1180-1202, 2010.
[40] R. Mur-Artal, R. J. M. Montiel, J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot*, vol. 31, pp. 1147-1163, 2015.

Copyright © 2021 by the authors. This is an open access article distributed under the Creative Commons Attribution License (CC BY-NC-ND 4.0), which permits use, distribution and reproduction in any medium, provided that the article is properly cited, the use is non-commercial and no modifications or adaptations are made.

Rami A. AL-Jarrah is Assistant Professor at Mechanical Engineering Department, Hashemite University, Jordan. He graduated with Bachelor and Master Degree from Jordan University of Science and Technology in 2002 and 2004, respectively. He received Dr.-Ing. Of Control and Robotics Systems from Regelungs-und-Steuerungstechnik Institute, Siegen University, Germany 2015.

His research interests are Mechanical Design, Fuzzy sets, Fuzzy Control, Mobile Robots Navigations and Optimizations, Computer Vision applications as well as Wireless Sensor Networks. He has a membership of the Jordanian Engineering Association and has been actively participated in several international research conferences since 2013.

Dr.-Ing. Rami AL-Jarrah has various publications including ISI and Scopus from the period 2013 to 2021. He has reviewed lots of research papers for international journals and has been acknowledged for the services.