Results in Engineering 18 (2023) 101130

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Addressing practical challenge of using autopilot drone for asphalt surface monitoring: Road detection, segmentation, and following

Hormazd Ranjbar[a], Perry Forsythe[b], Alireza Ahmadian Fard Fini[b], Mojtaba Maghrebi[c,*], Travis S. Waller[d]

[a] Department of Computer Engineering, Ferdowsi University of Mashhad, Iran
[b] School of Built Environment, University of Technology Sydney, Australia
[c] Department of Civil Engineering, Ferdowsi University of Mashhad, Iran
[d] Faculty of Transport and Traffic Sciences, Technische Universität Dresden, Germany

ARTICLE INFO

Keywords:
Road monitoring
Asphalt surface monitoring
UAV
Autopilot drone
Road segmentation
Road following

ABSTRACT

According to the recent world bank report, around 80% of a life-cycle cost of a road is devoted to maintenance which includes monitoring and repair processes. To more effectively keep road serviceability, knowing the current status of road health is crucial. Typically, the monitoring processes are handled by human-intensive procedures. So, automating this task could lead to saving time and cost and also, improve efficiency. Although, this task has been optimized by Ground Vehicles further. Yet it lacks the disadvantages of human-intensive procedure. As they are still semi-manual, creates traffic issues, and not being eco-friendly or cost efficient. In recent years, UAVs have been successfully utilized to handle a wide range of labor-based tasks including road assessments. This paper presents a drone-based solution to automate road monitoring and segmentation as well as addressing the practical challenges of using drones for this purpose. To do so, a platform is developed that controls a drone through a road monitoring flight using computer vision-based techniques. The platform, rather than sending maneuvering commands to the drone during a flight starting from takeoff to landing, firstly could detect road boundaries by finding vanishing points, and secondly, could identify the dash lines and the center of the road. Finally, the captured road is segmented and labeled with the temporal and geographical information supplied by the Inertial Measurement Unit (IMU) of the drone. It has been tried to optimize the platform in order to handle all the processes in real-time while the UAV is following the road during a flight. To evaluate the proposed idea, the developed platform is tested in urban areas. The achieved results demonstrate how effectively could detect and segment a road in different environments using an off-the-shelf UAV. This platform could improve the automation of the data gathering process required in road maintenance.

1. Introduction

Roads are one of the most important fundamental facilities in the world which are built to facilitate transportation and supply chain. This shows that roads are critical for economic development and their health needs to be assured by maintaining them in good condition. Road health conditions could bring immediate and sometimes dramatic benefits to users through better access to hospitals, schools, and markets. In contrast, bad road conditions could affect transportation capacity which could increase the travel time and cost. Moreover, Road condition shows an important aspect of the development of a country and its economic level and it has been adopted as rating criteria by the World Bank [1]. Thus, transportation agencies constantly make sure that they are doing appropriate maintenance planning decisions for road health and monitoring [2,3]. Hence, to keep the road in service, it's regularly maintained.

Road maintenance is a periodic task that requires construction repairs and monitoring to maintain roads in a good health. But road maintenance is a costly task, as its been reviewed in Ref. [4] this periodic work activity allocates about $500-$700 per kilometer per year. Furthermore, delaying road maintenance task causes high direct and indirect costs but if road defects repair promptly, they will cost less. As the *South African National Road Agency Ltd.* estimates, repairing costs rise to six times maintenance costs after three years of neglect and to 18

[*] Corresponding author.
E-mail address: mojtabamaghrebi@um.ac.ir (M. Maghrebi).

https://doi.org/10.1016/j.rineng.2023.101130
Received 31 August 2022; Received in revised form 25 January 2023; Accepted 26 April 2023
Available online 2 May 2023

times after five years of neglect [4]. Due to the labor-intensive procedure that road maintenance has, an enormous amount of money debit is issued that obliges many countries to spend only 1/5 to 1/2 of what they should initially spend on road maintenance of their own road networks [4].

Although there has been some type of work in this field of research, most of them try to satisfy the problem with equipment that costs as much as maintaining it labor-based (e.g. ground vehicles with onboard sensors) [5,6]. But both of these mentioned researches [5,6] are currently based on manned ground vehicles that are not the best-suited device to do this task perfectly and cost-effective. Since they have challenges such as environmental pollution and the need of driving vehicles.

Hopefully, in recent years, autonomous navigation systems have assisted a wide range of industries to establish more reliable, efficient, and faster procedures [7–21].

In addition to the adaptation of UAVs in industries, road monitoring is another field in which UAVs can be utilized on. With the help of UAVs, road infrastructures can have a significant improvement in time efficiency and reduce human intervention for monitoring road health. Therefore, UAVs could be hired to find road distress and segment road very easily by flying in mid-altitude or low-altitude areas. Doing this task with an economic drone helps to reduce the cost of maintaining outland or crowded roads. On top of that in terms of being cost-effective, it could assist authorities to save money allocated for road maintenance tasks.

In this paper, a platform is developed to automatize and tackle the road monitoring task. This platform consists of an out-of-shelf UAV and an application to communicate with a drone by giving it commands. The platform starts with road detection by using geometry features and vanishing points. Furthermore, a road following technique is developed for the drone. Meanwhile, the application of the drone tries to segment the road based on the dashed line on the road whether it is detected or not. Finally, the segmented road is attached with the temporal and geographical information supplied by the inertial measurement unit of the drone. This procedure of the platform should be able to operate in real-time even in high-quality video streams. It is being tried to do this task with an out-of-shelf drone which among the other UAV is in the mid-range class. All of these processes altogether could provide a reliable chance that makes a unique system for road surface monitoring with UAVs which also is a cost-effective solution.

In the rest of the paper, related works are presented in section II and then there is methodology in section III, and finally, experimental results and conclusion are presented in sections IV and V.

2. Background and related work

Automating the road monitoring task can be separated into two procedures that's has been widely adopted:

i) road detection
ii) road following

Road Detection means segmenting a targeted road from a given image. *Road following* can be the next step for autonomous road monitoring. It's a collection of maneuvering commands to trace the road within self-navigation. Table 1, summarized the picked approaches of the most related works of road detection. As it can be perceived from Tables 1 and in this road detection aspect mostly the introduced papers are focused on Appearance Cue based methods and learning-based methods.

Table 1
Overview of the most related works focused on Road Detection.

Road Detection Technique	Reference
Appearance Cue	Kong, 2010 [22]
Appearance Cue	Kühnl, 2011 [23]
Convolutional neural networks (CNN)	Alvarez, 2012 [24]
CNN	Laddha, 2016 [25]
Appearance Cue (Color Based)	Wang, 2014 [26]
Deep CNN and LiDAR	Chen, 2019 [27]

In the following sections, the most related works in road detection and road following are reviewed in detail.

2.1. Road detection

Road detection approaches are used vastly in other platforms such as Advanced Driver Assistance Systems (ADAS) [28], and road network detection as a utility [29]. These approaches can be grouped into two factors*: i) the altitude* of the observer device, and ii) the *road structure*.

The distance to the road imagery is based on the altitude of the capturing devices to the surface of the ground (referencing point) which creates different approaches for road detection techniques. Based on altitude, these devices can be categorized into three methods: *i) satellites* as the highest-altitude imagery robots [29]. *ii) Ground Vehicles* that have the lowest referencing point [22], *iii) UAVs* either Modern drones (known as copter drones) or Wing copter drones which perform in mid and high altitudes [30]. From these three mentioned methods, monitoring roads at high altitudes (e.g. satellite images, aerial images) is useless due to the loss of quality of captured road images and being a highly expansive method. However, the reason for this imaging is to identify the road network for Geographic Information System, [31–33]. For instance, Mokhtarzade et al. [29] developed an Artificial Neural Network (ANN) for road network detection to extinguish road pixels from non-road pixels. Now the hardware technology proceeded, satellites have been equipped with multispectral channel sensors such as Quickbird, IKONOS, and SPOT-5, which provided richer spatial information. Therefore some of the methods have been utilized on these sensors [34,35]. Another type of road detection approach is for the low-altitude referencing point mostly used in ground vehicles with an onboard camera. In this referencing point, most approaches are based on color cue (texture) detection [36–38], geometry features [39–43], a combination of road color and geometry features (like road boundary information) [36], and machine learning-based approaches for road detection [44]. One of the machine learning methods that have gained popularity owing to its accuracy is deep learning approaches [24,25,27, 44–46]. Where it can be more commonly seen in Semantic Segmentation, and Image Classification field where methods are trained on urban datum [47]. Even though this approach outperforms other road detection approaches in accuracy but still the lack of a specific dataset (road images in mid-range altitude) for UAVs and its limitation in processing power at the moment, makes using this approach useless. Besides, a fine-tuned CNN that have been trained for road detection in ground vehicles cannot be used to detect road images that have been captured by UAVs at higher altitude. It's because of the unrelated dataset that the CNN was trained on. Although these methods mostly are used in monocular cameras because of their beneficial price, but in some studies, it's worth mentioning that laser [48], radar [27,49], and stereovision [50] concepts have been used for road detection. Whereas monocular cameras, could be a better choice owing to their low-cost price and low energy consumption.

The second factor of road detection was road structure causes different approaches for road detection it can be classified into two classes: *i) structured roads* (e.g., urban roads) and *ii) unstructured ones* (e.g., dirt roads). In structured road detection also called lane detection, the most common approaches are Color cues [36–38], which use segmentation of road based on the color of the road. Alongside color cue approaches there are Hough Line transform [39,40], Steerable filters [51, 52], and Spline model [41–43] approaches which work with spatial features of the road to detect it. They have efficient computation time which makes them suitable for real-time approaches. Alternatively, for unstructured roads or roads without remarkable boundaries Alon *et al* [53], have combined the Adaboost-based region segmentation and the boundary detection constrained by geometric projection to find the “drivable” road area. However, it needs different types of road images to train a region classifier, which might be onerous in gathering and labeling the data. Kong et al. [22], created an algorithm named Locally Adaptive Soft-Voting (LASV) based upon road vanishing point estimation. This algorithm uses two dominant edges for the segmentation of the road area.

2.2. Road following

Road following task for UAV means following the path of the road while trying to centralize the drone with the road. During the navigation, segmentation should be done along with navigation. Rathinam et al. [54] proposed a method for river following using a fixed-wing UAV flying at 20 m/s with an onboard monocular camera. Similarly, Rathinam et al. [55] addressed linear structure followed by UAV which they followed a canal in their experiment. Frew et al. [56] proposed a road following in a wing-copter drone that tracks the roadway to determine UAV flight motion. They verified their strategies by hardware in loop (HIL) simulation over several kilometers of a straight line. Also, they did a hardware experiment with a successful road following an airfield runway. They aimed to present a control strategy for Wing copter UAVs to follow the road track by using only the vision-based methods.

In this study, the road detection approach tries to find the road boundaries automatically without any human interference. While keeping the minimum computation power to make real-time detection as the aircraft flies over the road. Hence, a new road detection for UAVs by using the main idea of [22] is introduced to detect the subject road with a single image without any previous data and in real-time with minimum computation. The authors in Ref. [22] presented a road detection method with adequate computation power in real-time scenarios for Ground vehicles by finding the perpendicular vanishing point. But instead of multiple Gabor filters for finding the vanishing point. In this proposed method, an enhanced method for vanishing point estimation based on [57] with several tweaks is used for road detection. Which makes the algorithm to require less computation power and faster.

3. Proposed methodology

In this paper, a platform is developed to automatically segment a targeted road surface captured by non-expensive UAVs. The proposed platform could be used for practical applications such as road monitoring and infrastructure health assessment. This platform consists of low complexity and, high-performance computer vision algorithms which are in charge of three main modules (Fig. 1).

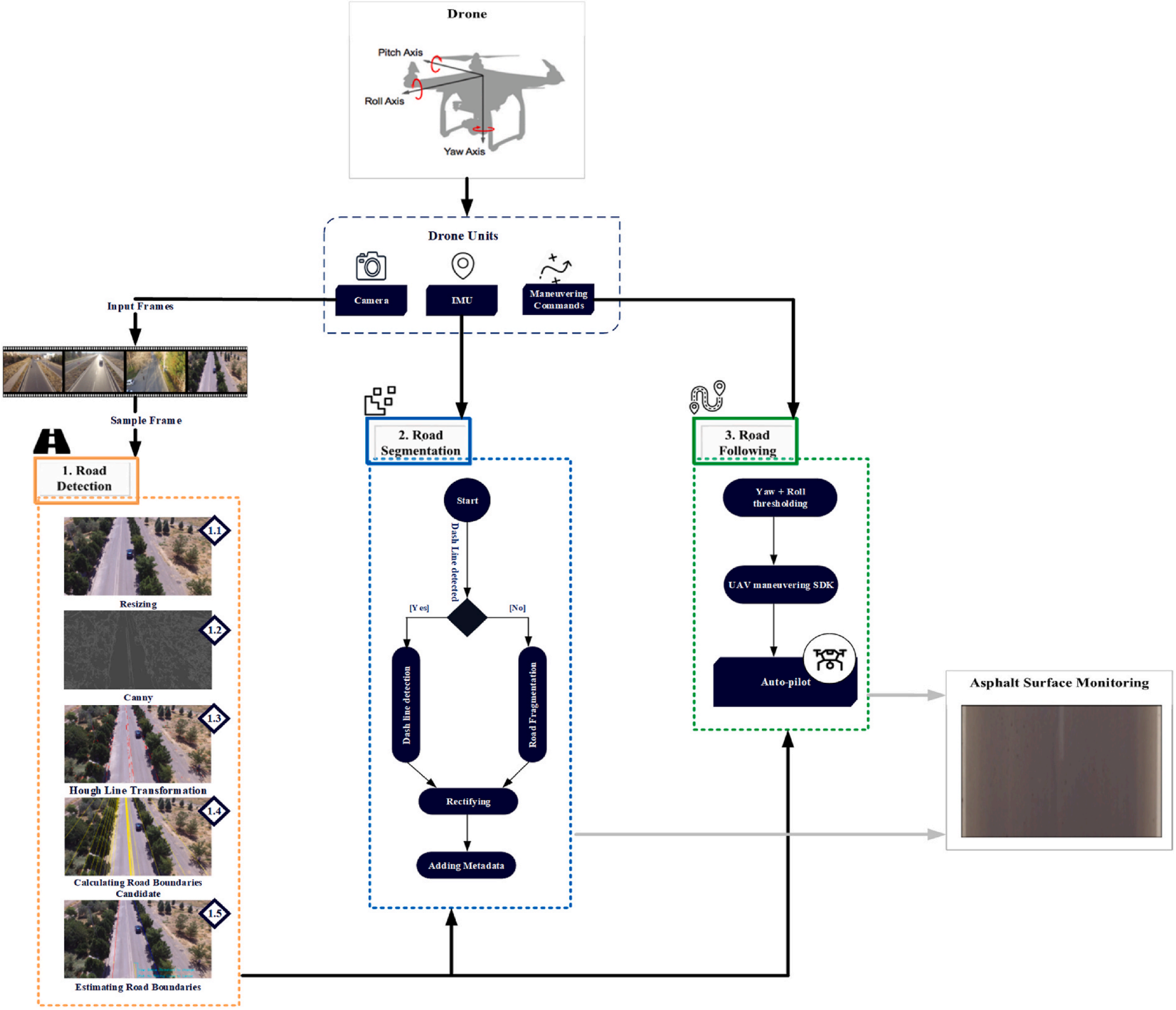


Fig. 1. The platform overview consists of three modules in the application, Road detection, Road segmentation, and Road following.

(a) Road detection
(b) Road segmentation and rectification
(c) Road following

The platform uses the UAV camera to detect the road boundaries and segmentize the captured road while the aircraft is flying. The GPS sensor of the aircraft embedded in the Inertial Measurement Unit (IMU) is used to obtain geographical coordination which later will be used to label each road segment. The entire tasks related to the drone are controlled by the platform automatically without human interference. This autopilot feature of the developed platform provides a chance to reduce human costs and avoid human errors. Auto-pilot module of the developed platform is in charge of all flight commands (excluding take-off) as well as flight safety issues to make sure that the aircraft stays at the center of the road while it's moving forward. Also, to be noted the methodology assumes the road that is going to be inspected is a straight road without curved angles.

The platform needs a routine initialization including locating the drone close to a road, connecting the drone to the control unit, and take-off. After platform initialization, the aircraft starts to detect the road by finding the road boundaries with the help of the perpendicular vanishing point. Then, the detected road frame is segmented and labeled with the 2D position of the aircraft (including latitude, and longitude) for possible further maintenance works. In parallel with road segmentation, the road following module that consists of the auto-pilot module of the platform is in charge of making sure that the drone always flies along with the center of the road. This module also corrects the drone flying path if wind deviates the aircraft.

The tangible outcome of the developed platform is a number of captured images labeled with spatial and temporal information. The acquired images can be utilized to diagnose the health condition of each part of the road.

In the following sections, the three main modules of the platform are described in detail.

3.1. Road detection

Assuming the aircraft is deployed successfully at the point of the subject road at midrange altitude the first step is to find the vertical vanishing point of the image. If the vanishing point *vp* is the hypothesis vertical vanishing point of the image there is a set of l_i that are all the edge lines intersect with the *vp*. These lines are all parallel to each other in the real world. It can be assumed that the road is between one of those two parallel lines. So, in l_i set, there is at least a combination of two lines which are the right and left boundary of the image frame road. With this assumption, the algorithm starts to estimate the *vp*.

3.2. Vanishing point estimation

To begin the vanishing point estimation there are a couple of pre-processing steps that need to be performed. First, the connected device receives the image frame of the streaming video from the drone as a raw input. Then, size reduction is applied in order to increase the performance and promote real-time ability in weaker devices (portable devices like mobile). After size reduction, the frame consists three-dimensional array with $(1280, 720, 3)$ size where the first, and second dimensions are the width and height of the frame respectively, and the third dimension is the color channel of the frame.

After that, histogram equalization (HE) and the canny edge detector algorithm [58] with a default sigma value of $\sigma = 3$ are applied respectively for detecting the edges in the given frame of the video. For HE method the cumulative histogram equalization is used [58]:

$$HE(v) = round\left(\frac{cdf(v) - cdf_{min}}{(M \times N) - cdf_{min}} \times (L-1)\right) \tag{1}$$

Where v is the pixel intensity in the frame and cdf_{min} is the minimum value of the cumulative distribution function, $M \times N$ gives the image's size and L is the number of image grayscale levels.

The resulted image from HE equation and canny edge detection is used in a probabilistic approach of Hough Line transform [59] to detect straight lines of the frame. Extracted lines are filtered by minimum line height and maximum gap point based on the image size. The minimum line height and maximum gap are calculated as follows where M and N are the width and height of the raw input image frame respectively. p and e are the static ratios: (both p and e in our experiments were set to be 0.006 and 0.0016 respectively.)

$$min\ Length = \left\lfloor \sqrt{M^2 \times N^2} \times \mathrm{p} \right\rfloor \tag{2}$$

$$maxGap = \left\lfloor \sqrt{M^2 \times N^2} \times \mathrm{e} \right\rfloor \tag{3}$$

This thresholding gives an advantage over flexing the algorithm to accept the different sizes of an image without manually setting new values for each. After the thresholding, to avoid unwanted noise that is made by the objects beside the road i.e. grass, trees, or buildings, the angle θ between each line and the vertical axis of the image is calculated should be $\theta > 25^\circ$ or it is discarded and not used in vertical vanishing point detection. Every resulting edgelet can be represented as $E = \{\overrightarrow{x}, \overrightarrow{d}, s\}$ where $\overrightarrow{x}$ represents the edge line coordinates, $\overrightarrow{d}$ represents the edge direction (gotten from the eigen value of the covariance matrix), s is the edge strength. In short, an *edgelet line*, $\overrightarrow{l_E}$, corresponds to an edgelet E where the line passes through $\overrightarrow{x}$ and it's parallel to $\overrightarrow{d}$. These edgelets are sorted in an array, *edgelet array*, by their edge strength in descending way. Then, RANSAC [60] based Vanishing Point algorithm detection is used for finding the vanishing point in the frame of the image with the help of the edgelets.

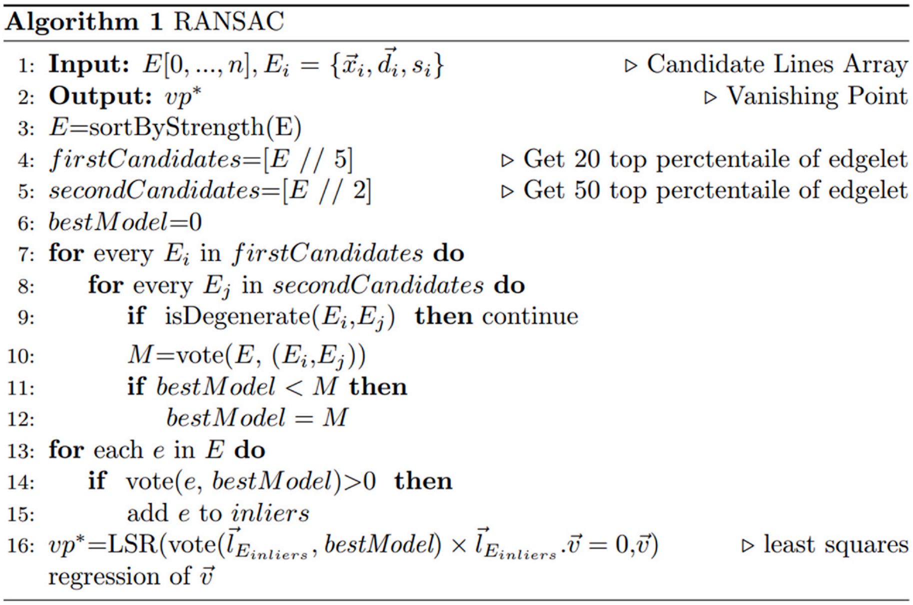
The algorithm that is used to detect vertical vanishing points is similar to the one in the Auto-rectification of user photos [57] algorithm. The RANSAC algorithm uses random pair of edgelets E to calculate the vanishing point. Every pair of edgelet $E_1(\overrightarrow{x}_1, \overrightarrow{d}_1, s_1), E_2(\overrightarrow{x}_2, \overrightarrow{d}_2, s_2)$ can be modeled as $M(E_1, E_2)$ which makes the hypothesis vanishing point that is the intersection of the two edgelet lines. The intersection of the model M can be calculated as a cross-product of the two edgelet lines: $vp_{(E_1,E_2)} = \overrightarrow{l}_{E1} \times \overrightarrow{l}_{E2}$. The hypothesis vanishing point can always be calculated except in the cases where the random pair of edgelets are the same. These exception pairs are dismissed in the calculation. To achieve better performance the edgelet pairs aren't selected completely randomly. Instead, the first edgelet, E_1, is selected from the top 20 percentile of the edgelet array (mentioned above) and the second edgelet, E_2, is selected from top 50 percentile [57]. This effectively biases the algorithm towards stronger edges.

After finding hypothesis vanishing point $vp_{(E_1,E_2)}$ for every model $M(E_1, E_2)$ the algorithm iterates over all other edgelets $E_i(\overrightarrow{x}_i, \overrightarrow{d}_i, s_i)$. Each E_i casts a vote for model M as follows [57]:

$$vote(E_i, M(E_1, E_2)) = \begin{cases} \dfrac{1 - e^{-\lambda \cos^2 \theta}}{1 - e^{-\lambda}} & if\ \theta \leq 5^\circ \\ 0 & otherwise \end{cases} \tag{4}$$

Where θ is the smaller angle between $\overrightarrow{l}_{E_i}$ and the line consisting of two points $\overrightarrow{x_i}$ and the hypothesis vanishing point from model M and also λ is a system parameter. This voting system gives the maximum vote when the edgelet E_i and model M have $\theta = 0$ and the voting rate decreases when θ goes to 90° but it has a threshold that votes to zero when the $\theta > 5^\circ$. The model M_i with the maximal vote is the estimated vanishing point vp_e. For better insight there is an algorithm overview on Algorithm. 1.

Algorithm. 1
Vanishing point estimation algorithm



```
Algorithm 1 RANSAC
Input: $E[0,...,n]$, $E_i = \{\vec{x}_i, \vec{d}_i, s_i\}$        ▷ Candidate Lines Array
Output: $vp^*$                                                  ▷ Vanishing Point
$E$=sortByStrength(E)
$firstCandidates$=[E // 5]                  ▷ Get 20 top perctentaile of edgelet
$secondCandidates$=[E // 2]                 ▷ Get 50 top perctentaile of edgelet
$bestModel$=0
for every $E_i$ in $firstCandidates$ do
    for every $E_j$ in $secondCandidates$ do
        if isDegenerate($E_i$,$E_j$) then continue
        $M$=vote($E$, ($E_i$,$E_j$))
        if $bestModel < M$ then
            $bestModel = M$
for each $e$ in $E$ do
    if vote($e$, $bestModel$)>0 then
        add $e$ to $inliers$
$vp^*$=LSR(vote($\vec{l}_{E_{inliers}}$, $bestModel$) × $\vec{l}_{E_{inliers}}.\vec{v} = 0$,$\vec{v}$)        ▷ least squares
regression of $\vec{v}$
```

Once the best model is identified, a re-estimation is done to achieve a more accurate vanishing point. For the estimated vanishing point, vp_e, we find the inlier edgelet lines corresponding to, the vp_e with $S = \{E_i \mid vote(E_i, M_{best})\}\rangle 0$ this makes a list of inlier edgelets. For finding the optimal vp^*, ideally the edgelet line $\vec{l}_{E_i}$ from set of S should pass through vp^* point. In homogenous mathematics, it can be achieved with dot production as $\vec{l}_{E_i}.vp^* = 0$ also this formula can be more efficient by adding weight for each inlier by having its vote into dot production which makes [57]:

$$w_i \times \vec{l}_{E_i}.vp^* = 0 \tag{5}$$

Having (5) solved in the least squares regression gives out the optimal vanishing point only with the defined inlier edgelets. This leads to the final calculation of vanishing point vp^*.

3.3. Road boundary detection

After estimating the vanishing point vp^*, Road boundary detection begins. In the detected edgelets E_i only the ones from set of S are included for the road boundaries criteria. Furthermore, every computed edgelet line, $\{E_1, E_2, \ldots, E_n\}, n \geq 2$, is assigned to the same group set if they have $\left|\vec{d_i} - \vec{d_j}\right| < 5^\circ$. In every set only the edgelet that is closer to the bottom of the frame is selected as the candidate of the set because the bottom edgelets have better image quality and its more noise prone. This work reduces the edgelet candidates which help in better performance optimization (see Fig. 2).

Then, each parallel line in the real world intersects with the vanishing point of the frame, and a pair of these candidates are able to be the boundaries of the frame road. Also, for these candidates each edgelet E_i with endpoint e_i is calculated as $l_{E_i} = (x_i, N)$ where N is the height of the frame (Meaning that the x_i is the horizontal coordinate where the line l_{E_i} intersects with the vertical bottom of the frame) After calculating the candidate x_is, those points that fall out of $-0.3 \times M < x_i < M \times 1.3, e_i = [x_i, N]$ threshold where M is the width of the frame are discarded. In

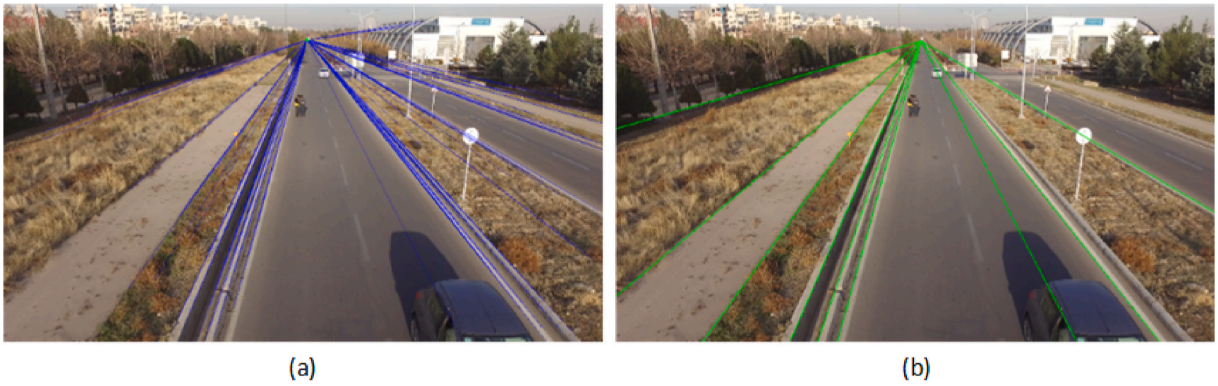


Fig. 2. Reducing line candidates. a) is before thresholding b) is after thresholding.

other words, those edgelets whose endpoints are positioned more than 30% (this ratio is calculated by the width of the current frame) outer scope of the frame will be deleted. This filtering can be done because this study assumes that UAV is hovered on the road hence the road boundaries never exceeds the UAV camera angle of view.

Next, road competence measurement is applied. Each pair of candidate areas is measured with edge pixel density. Before, measuring the authenticity of road boundaries. The distance between the candidate pair's endpoint needs to be more than 40% of the width of the image. Otherwise, the candidate pair will be discarded.

Now that every pair creates a segmented area, for every pixel in that area an image I is formed by with sets of pixels $I = \{u_{xy}, (x,y)\varepsilon\, \beta\}$ and $\beta \subset R^2$ which every u_{xy} is a binary value of being edge or not labeled as $u_{xy} \in \{0,1\}$. u_{xy} is assigned to the pixel at the position (x,y) of the area. If the quantity in I pixels is shown by n, then edge pixel density is calculated as (6). Eventually, the road boundary is selected by calculating m for every combination of two candidate lines i and getting the minimum i pair candidate as shown in (7).

$$m = \frac{\sum u_{xy}}{n} \tag{6}$$

$$b = arg_i\ min\left(\frac{\sum_{xy}^{i} u_{xy}}{n_i}\right) \tag{7}$$

3.4. Road rectification and road segmentation

After finding the road region in the frame, the detected road can be segmented for further inspections that it's used in the road monitoring procedure. These segments need to be added with temporal and spatial metadata of their own to ensure that when and where distress or other abnormalities are found in one of the road segments. To do so, framing the whole road region is unnecessary because in road monitoring as the drone moves forward only the lowest portion of the road can be truly labeled with accurate metadata and image quality which comes from the aircraft IMU and camera.

In our study, based on the road characteristics there are two methods for road region segmentation: *1)* cropping a portion of the frame while the drone moves forward in a linear speed and *2)* dash line tracking which means cropping the frame based on the dashed line marking of the road.

When there are no dash lines detected in the road, the algorithm crops p percent of the image height from the bottom in every f frame (these two parameters depend on the speed and distance that UAV can travel in every second) as shown for an illustration in Fig. 3. (In tested experiments 15% portion of the road region and 5 frames per segment was completely fit).

If the road has dash line marking in its region, segmentation is done by dash line tracking to the targeted road. In-dash line segmentation, the road is segmented by the detected dash line marks in the road region. As the road boundaries are known to the system, the dash line segmentation is applied only on the road mask to reduce detection error of finding the dash lines, then finding contours with outer retrieval and simple approx. the approach is used [61]. For removing noise objects and saving in memory management, contours that areas are less than 10 pixels are ignored. Then, The Minimum Bounding Rectangle (MBR) is applied to the contours. Now, each valid contour is presented by a rectangle which is a dashed line. The center of these dash lines is calculated by their up-left and down-right points as shown below:

$$centroid_i = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}\right) \tag{8}$$

Next, these dash line rectangles contours are sorted by y-axis with their centroid.

Now in every frame, each rectangle is tracked. In the first frame, the algorithm sets all centroid point to a unique ID, then in the next frame it tries to update centroids by finding their new location, this helps in finding new frame centroids and checking the nearest centroid to the old one. If there was a distance below a certain threshold (in these experiments 50 pixels as distance threshold worked decently) the centroids are updated to their new location, after that all unlabeled new centroids are assigned as new centroids. The method uses a unique ID for every dash line object and updates their centroid. Meanwhile, if there is a dashed line that has not been seen in consecutive frames for about a threshold (in these experiments 3 consecutive frames worked decently), it is discarded.

After dash line tracking, each dashed line is labeled with a unique ID along with the position they are on. So, in that frame the labeled dash line can be shown as (Id, X, Y), and all of them are sorted by their Y's as their position in the vertical axis. The segmentation is applied when on the road the lowest dash lines (the lowest is the one that has the minimum Y amount) are dismissed from the video frame. When segmentation is triggered, the specific road region from the bottom to the next dashed line is captured. If the next dash line can't be found, p percent of image frame height is captured (In the experiments 15% portion is used). The p percentage of the image frame is user variable which is mostly based on the camera resolution, and user preferences.

Whether any of the two mentioned road segmentation approaches is applied to the road, the segmented road region is rectified to enhance the monitoring experience. Also, in every road region segmentation with the help of UAV IMU, the GPS coordinate of the drone and the captured region segment timestamp can be collected to be monitored for later use.

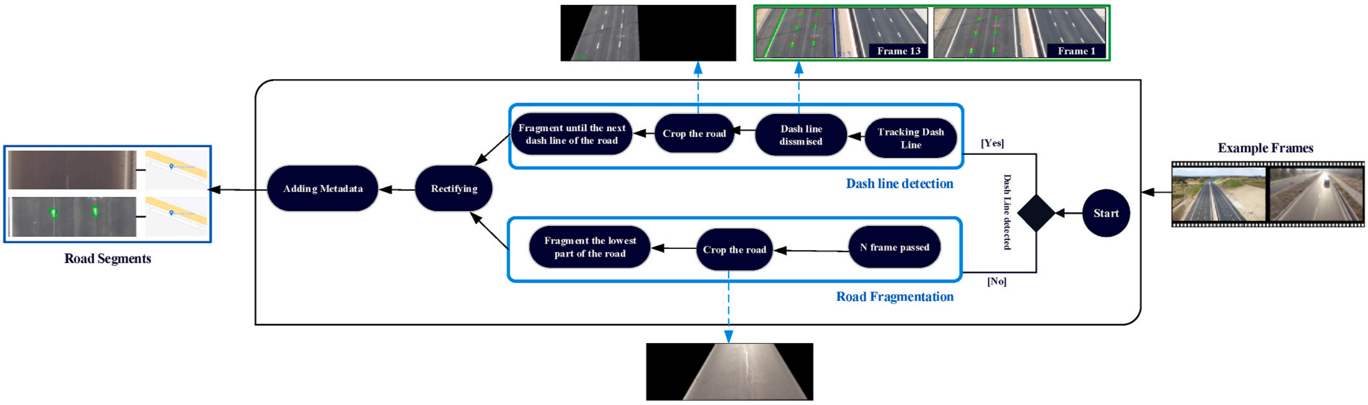


Fig. 3. Road segmentation and rectifying overview. This figure shows the two mentioned methodology procedures in road segmentation if the dashed line is detected.

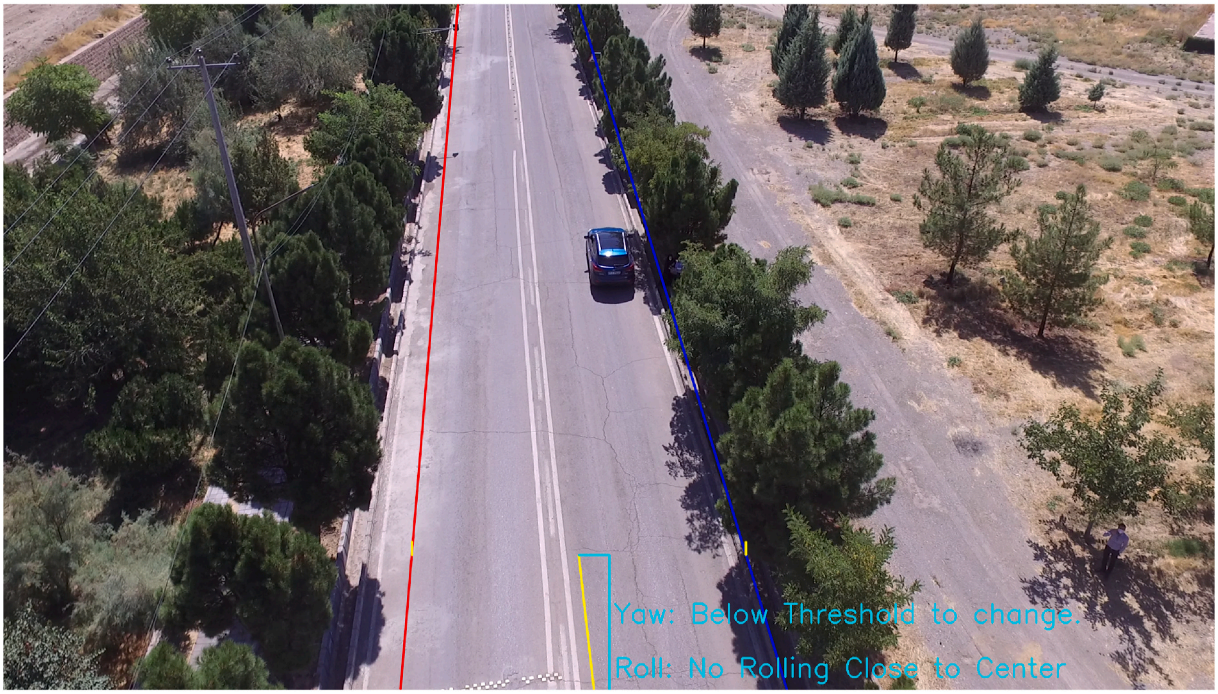

Yaw: Below Threshold to change.
Roll: No Rolling Close to Center

Fig. 4. Following the road by controlling Yaw and Roll of the drone.

3.5. Road following

While drone saves image segments of the road, sequentially to be able to do road segmentation, road following is needed as well in this system, thus after finding road boundaries, the road following technique is measured by using the centerline of the road boundaries. This center line is calculated by vanishing point and the center of the road boundaries end-points (Fig. 4. Illustrate this). Then, maneuvering commands are utilized to center the aircraft with the target road. Which makes the UAV to localize itself toward the center of the road.

By assuming that drone flight controls can be translated into lateral movement, forwardness, and rotation, the road following method gives relative instructions based on center line angle with horizon and distance of the center line to the center of the image. Here in our experiment as fully mentioned in section IV the tested drone is completely able to translate instructions with help of converting the speed into the duration of them to happen. As shown in the top-left of Fig. 1 the drone control system has Roll, Pitch, and Yaw values for movement. Which is equivalent of X, Y, and Z rotation on their axes respectively. In drone Flight handling, Pitch is used for heading the drone forward and backward and needs to be chosen correctly to prevent noise and distortion production. Roll for lateral moving and lastly Yaw to change the face of the drone clockwise or counterclockwise (rotation) so there is no angle with the road center line. Making drones centralized to the road is easier and better for detection as shown in Fig. 4. The drone at every segmentation goes forward with the help of pitch value and then uses the yaw and roll to centralize itself with the road. Yaw is changed when the center line of the road makes an angle with horizon more than a specific angle which is a user variable (in this study it was 20°) so centralizing of the road is done by rotating clockwise or counterclockwise. Roll movement used a similar method to yaw but with distance measurements. The method only uses roll when the distance of the center line is more than the specific threshold (in this study it was 10% of the frame width). The rolled amount is set based on the direction and distance of the center line to the center of the image. The roll movement is the measurement of angle, and velocity that will be giving to the aircraft SDK as command to tilt the spacecraft either to left or right which will be discussed more in Section IV. And other options for in-flight movement like the altitude of the drone never change in these experiments.

4. Experiments and results

This section describes the configuration and setup of the proposed method. Then, two different datasets are introduced. Next, the main experimental results are provided. Finally, the performance of the algorithms is estimated with the main metrics.

4.1. Hardware and software configuration

The UAV that is used in this research is a quadcopter camera drone that is used for overall aerial photography. These drones are quite the best known for environmental photography by their price level. Operating on low-altitude or mid-altitude to capture the best quality images. The model of the drone that is used in this research is a DJI Phantom 3 Standard. It's one of the most economical drones that can be purchased in the industry with a satisfactory camera. The camera of the drone can move in 2-axis and provide a front view. This camera is a 4 K resolution camera with stability gimbals to avoid blurriness and video noise reduction while flying. Also, for the research, an Android mobile device with Exynos 9810 CPU, 6 GB of ram, and Android 10 OS (Android Q) has been used as a wireless bridge to connect the mobile device to the UAVs' controller and give the aircraft flighting command through the UAV SDK command.

Developed code, UI, App, and test data are available as supplementary documents to this manuscript.

4.2. Experimental setup

To evaluate the platform that is created, a mobile application has been developed. This mobile application in order to command the UAV that it is used (DJI Phantom 3 Standard) the mobile application communicates with the drone by using DJI mobile SDK [62]. DJI SDK also has another platform such as Windows PC and an Onboard chip that enables the platform to control the drone's movement and camera of it. Although Desktop platform SDKs have more flexibility in coding and can have better performance, yet these abilities require a higher-end class of

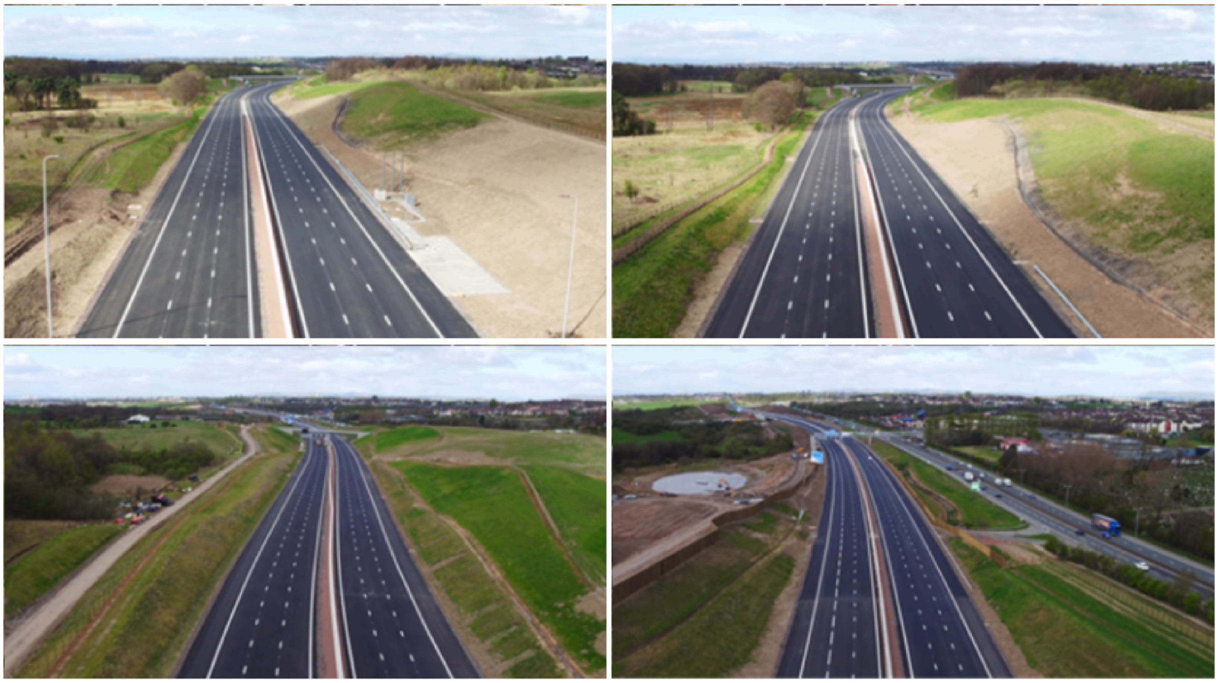


Fig. 5. First dataset samples.

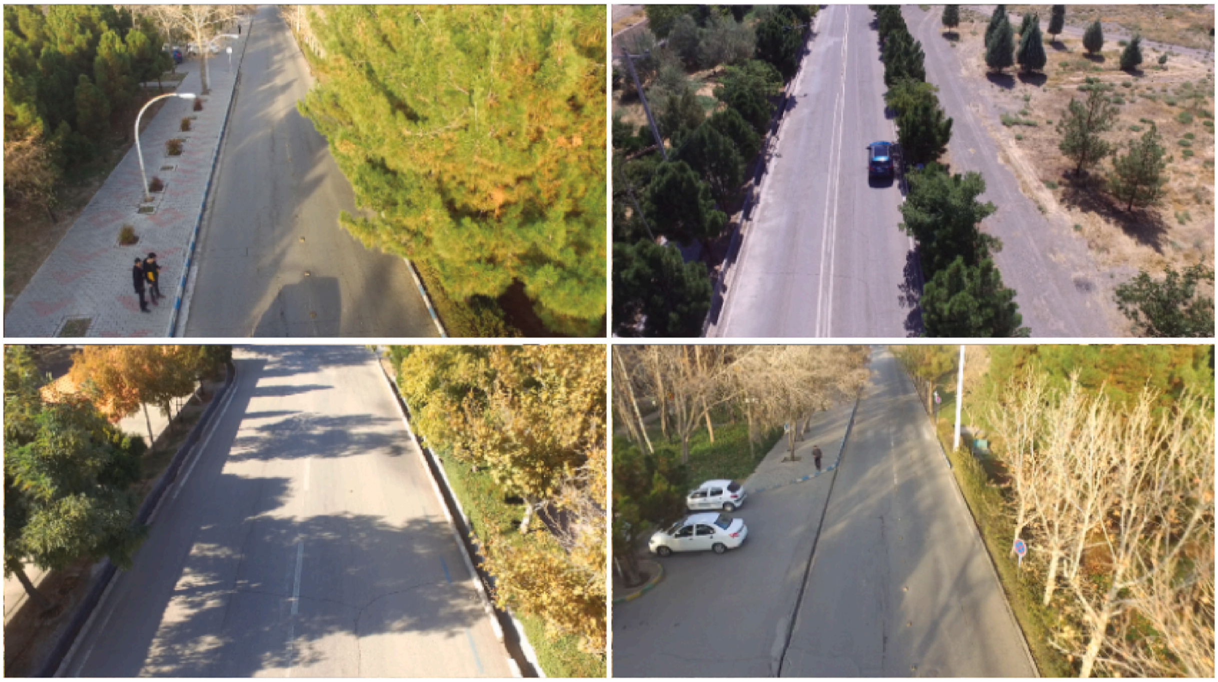


Fig. 6. Second dataset samples.

DJI aircraft which are more expensive and it lacks the mobility feature. For cost and mobility efficient approach mobile SDK is used. In mobile SDK two software environment has been used:

- The application is developed with Java as native android code and the code base interacts with DJI mobile SDK. The application transfers the flight commands and receives camera data and logs. Additionally, it's used to save road segments with the help of the drone Inertial Measurement Unit (IMU) to tag it along with GPS coordinates and the time.
- Python code is used for Machine Vision algorithm as its the fast way to implement these codes. This platform is integrated with the Android OS with using Chaquopy SDK which runs python codes in android environment. The python version used as implementation was 3.7. Also, the scikit-image library and OpenCV 4.0 (Open-Source Computer Vision Library) has been used in python.

4.3. Datasets

In Order to carry out the experiments, two datasets we gathered: 1) A

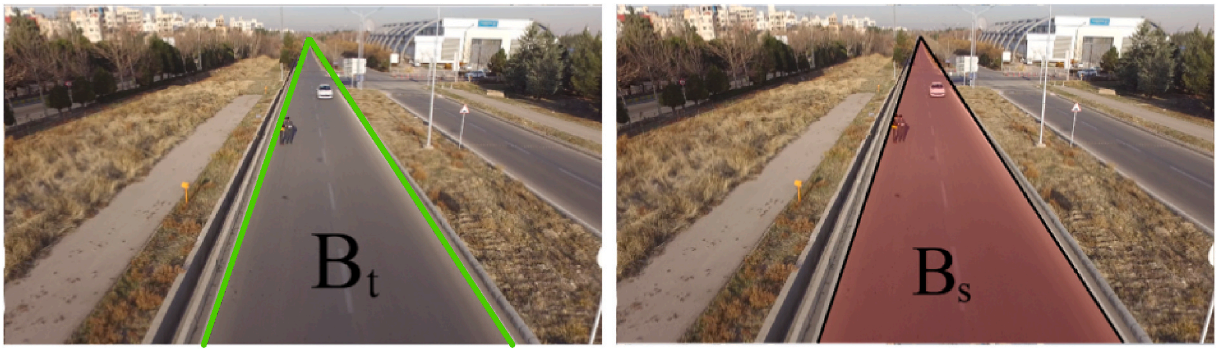


Fig. 7. This figure illustrates the road detection accuracy, Bt is the ground truth of the road area and Bs is the algorithm result of the road detected area.

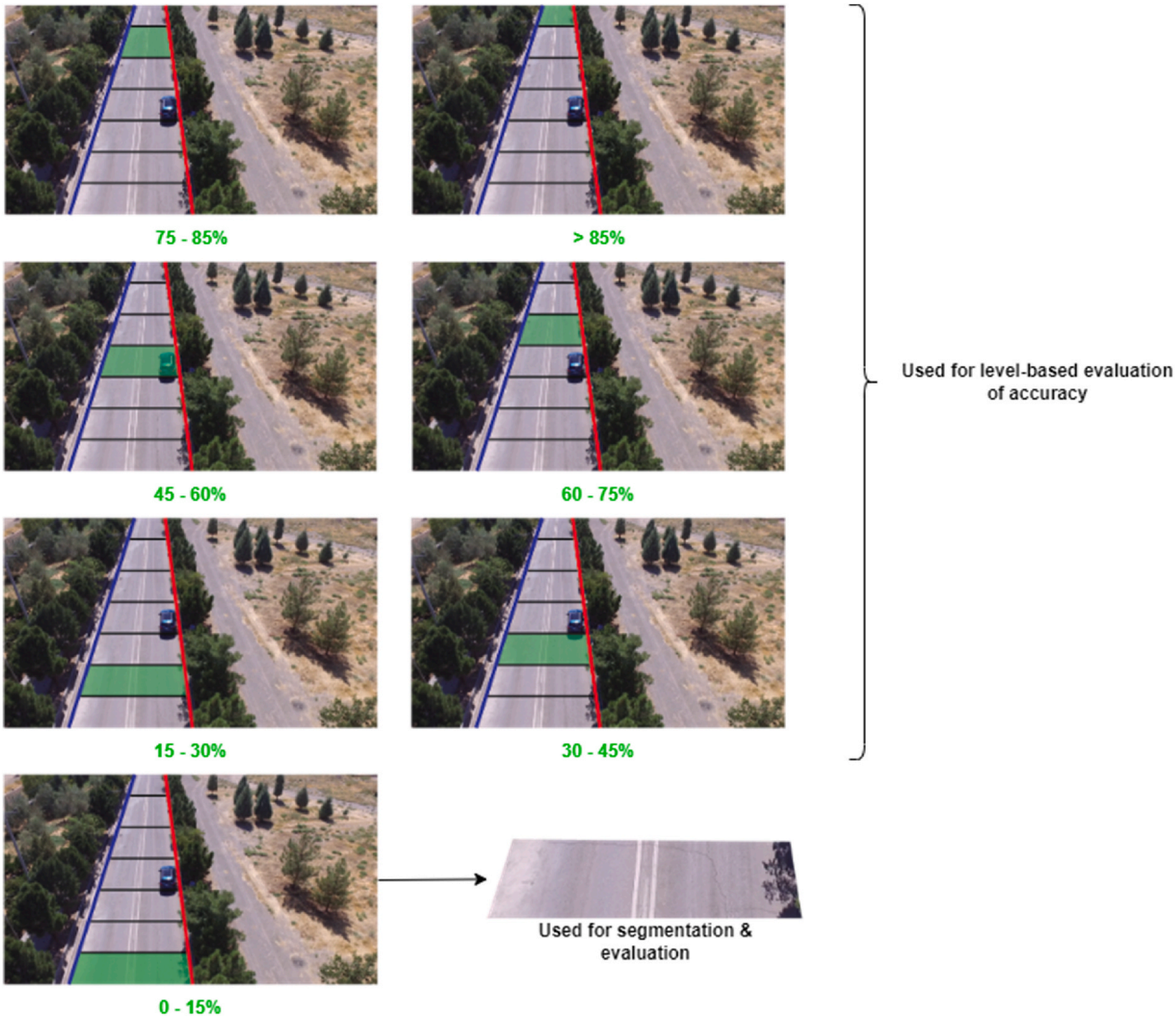

75 - 85%
> 85%
45 - 60%
60 - 75%
15 - 30%
30 - 45%
0 - 15%
Used for level-based evaluation of accuracy
Used for segmentation & evaluation

Fig. 8. An example of road divisoning for level based evaluation.

dataset from videos over the internet with 1920x1080 pixels quality (available as sup), 2) Video streams used in live tests from UAV drone, DJI Phantom 3 Standard, by using the 1/2.3 CMOS camera mounted on the drone. The second dataset is made while the drone was streaming a number of roads with variant illumination and surroundings at Ferdowsi University Campus roads. Dataset one is used only for road segmentation experiments and dataset two is used over the whole procedure of experiment process. The first dataset has more road lanes marking visibility using dashed line segmentation method, but the second one used a more practical style with no dashed line marking appearance. Figs. 5 and 6 are showing several sample frames in these two datasets. Out of the second dataset footage, 100 frames of it have been randomly exported for the experiment's evaluation.

4.4. Road detection measurement criteria

The accuracy of algorithms is mostly estimated by recall and precision concepts. The recall is computed as (9) and Precision is the ratio of

Table 2
Performance evaluation of the proposed detection algorithm based on the portion of the road on the second dataset.

Portion of the Road	Precision	Recall	F1-score
0–15%	93.48%	92.70%	94.23%
15–30%	96.57%	92.37%	94.06%
30–45%	96.67%	92.03%	93.87%
45–60%	96.71%	91.49%	93.54%
60–75%	96.53%	90.46%	92.80%
75–85%	95.87%	88.45%	91.26%
>85%	92.25%	77.39%	82.51%

Table 3
Time-consuming performance on the proposed methodology.

Image Size	Processing Time (sec.)
1280 × 720	0.35
800 × 600	0.26
640 × 480	0.21

true positive to whole retrieved detected road as its showed in (10).

$$recall = \frac{B_t \cap B_s\ (True\ Positive)}{B_t\ (Positive)} \tag{9}$$

$$precision = \frac{B_t \cap B_s\ (True\ Positive)}{B_s\ (Detected\ Road)} \tag{10}$$

In equations (9) and (10), as shown in Fig. 7, the ground truth is B_t and the detected road from the algorithm is B_s. To compute the performance of the road detection, F1-Score (F1) is used which is the harmonic mean of precision (Pr) and recall (Re) that it is mentioned above. The F1 equation is defined as below:

$$F1 = \frac{2 \times Pr \times Re}{Pe + Re} \tag{11}$$

As it was mentioned in the proposed methodology, road segments which are the final result of this study are a portion of the targeted road. Hence, the entire detected road accuracy is not necessary. That's why in road segmentation average accuracy is based on a portion of the road. As shown in Fig. 8, the performance evaluation in Table 2 is divided by the portion of the road. Also, to be noted in Fig. 8, the 6th road segment has 10% portion which is different to other divisions. This is simply due to the reason that the image frame isn't divisible to 15% and better road area distribution for accuracy evaluation.

As its shown in Table 2 the road detection algorithm shows an acceptable performance. While having adequate time processing performance on the whole proposed methodology including segmentation and following. Table 3 lists the running time on the proposed methodology based on the image size it gets as input on 8 core processors based on ARM architecture.

4.5. Vanishing point estimation

Measuring the correctness of the vanishing point is tested on 60 images from the second dataset. All images have been resized and questioned by three individuals to manually pick out the vanishing point. The overall ground truth vanishing point for the selected image is estimated to be the mean of the manually picked-up vanishing points. Hence, V_g is the 2D location of the vanishing point that is been asked and the V_s is the 2D location of the vanishing point that is estimated by the algorithm. Then the accuracy has been calculated for every *I* image in the dataset using (12). Considering V_g and V_s being in (x,y) sets $d(V_g, V_s)$ is the Euclidean distance of these two. Also, M and N are respectively the width and height of image *I*.

Table 4
This tables shows the Vanishing Point accuracy through the second dataset, it shows more than 75% of the images are below 1% of the distance between their threshold.

Difference of the Vanishing point in Percentage	Number of Images in the dataset
Below 1%	78%
Between 1 and 2%	4%
>2% and <3%	18%

Table 5
This table illustrates the average accuracy of the second dataset over different line lengths and maximum gap difference.

Minimum Length	Maximum Gap Distance	Road Detection Accuracy
25	10	84.478281
25	8	90.74258
25	5	93.424519
25	3	93.968856
20	10	81.710503
20	8	87.507017
20	5	90.04567
20	3	93.878717
15	10	88.648974
15	8	84.524158
15	5	89.314293
15	3	89.767359
10	10	76.879832
10	8	81.061006
10	5	86.221437
10	3	89.798193
5	10	84.86027
5	8	80.000824
5	5	89.970831
5	3	80.935054

$$V_i = \frac{d(V_g, V_s)}{\sqrt{M^2 + N^2}} \tag{12}$$

The accuracy of the vanishing point estimation is shown in Table 4.

For the performance measurements, the processing time in our specific android device is tested. The algorithm works in real-time (every frame takes 0.0625 ms average which is about 16–17 fps) in the mobile phone when it's connected to the UAV beside. To increase the efficiency some top percent of the frame can be ignored (like ignoring 10% top of the frames).

4.6. Probabilistic hough line transform thresholds

For getting the best results from our experiments in Hough line transforms, two factors are required: *i)* line length and *ii)* maximum line gap. Line length is tested from 5 to 30 and the maximum line gap from 3 to 10. This test ran on over 150 sample images on the second dataset with different urban road areas in color and illumination. The best line length and line gap were achieved was 0.6% and 0.16% dependent on the image diagonal size calculated in (13) which *W* is the width of the image and *H* is the height of it. Table 5 shows in different factors how road detection recall accuracy (9) behaves.

$$image\ size = \sqrt{W^2 + H^2} \tag{13}$$

4.7. Road following

In this study, the drone localization system that is been used is the body coordinate system. Which means the aircraft movement is dependent on the location and orientation of coordinate axes that is made and referenced by the drone itself. So, the perpendicular axes are defined such that the origin is the center of aircraft mass. As its been illustrated in Fig. 1, the X axis is directed through the front of the aircraft and the Y axis through the right of the aircraft (related to itself). Using

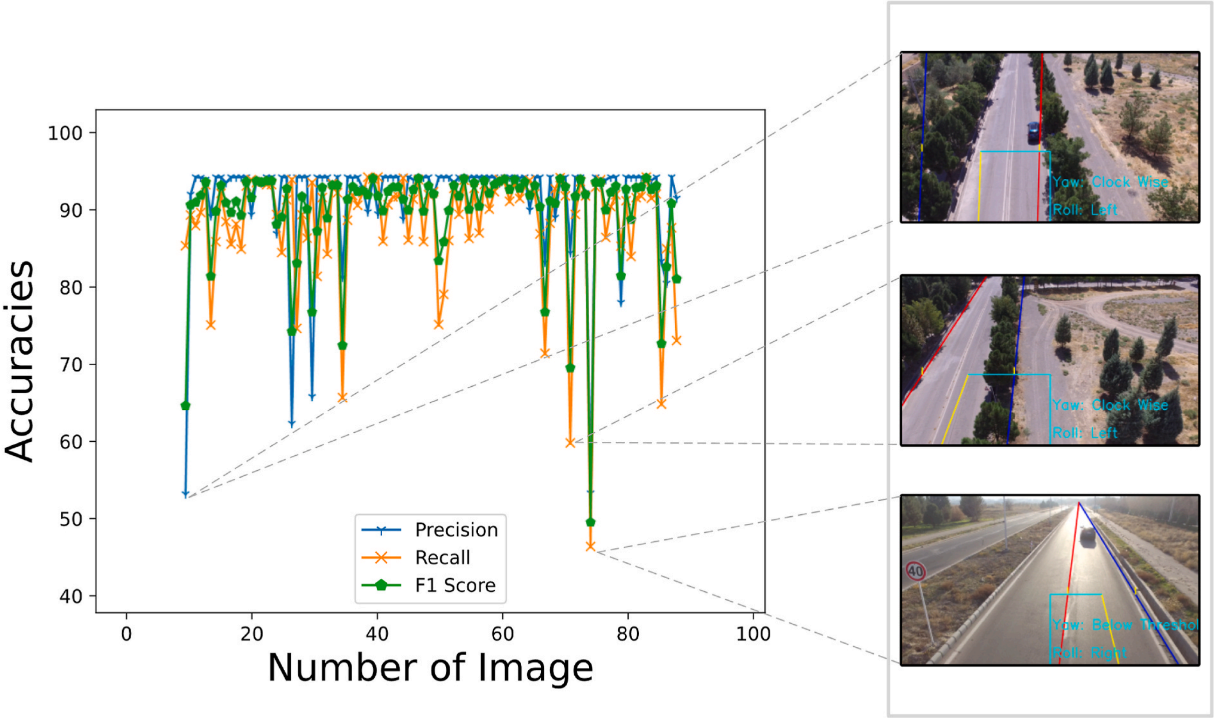


Fig. 9. Inaccuracies in finding the road boundaries. In the example images, road boundaries can't be estimated correctly because of the car object or trees presenting themselves as strong edges.

the coordinate right hand rule, the Z axis is then through the bottom of the aircraft. Now, each definition of rotational change with certain degree of angle while having velocity on the perpendicular X, Y, and Z axes creates the body coordinate system roll, pitch, and yaw respectively. Which further results in X, Y, and Z movement in these coordinates. This translation happens inside the DJI SDK as the rotation, and thrust (which are adjusted) makes the drone to move around these axes. For example, if the aircraft is commanded to go forward by a positive angle on Y axis (positive Pitch value) and thrust, the back propellors spin faster and have move thrust than the front propellors which creates the aircraft to go forward. The flight controller automatically balances the thrust on each propellor.

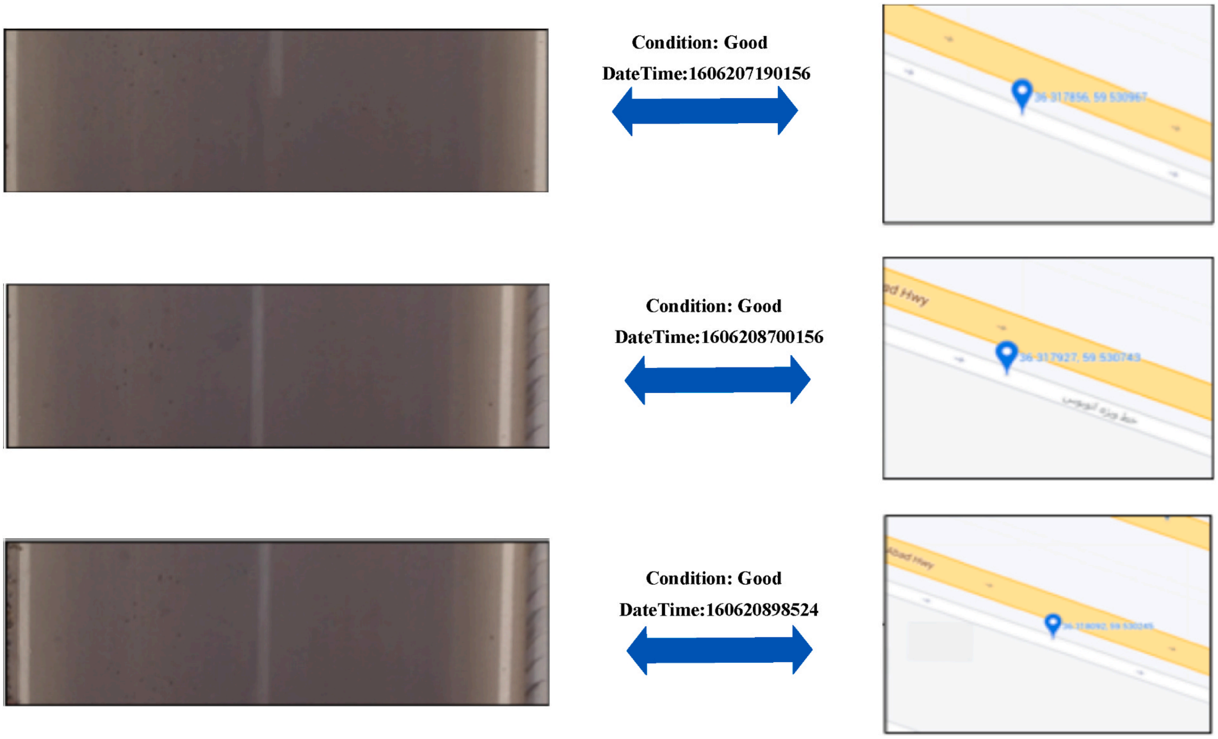


Fig. 10. Segmented road fragments with a spatial and temporal label ready to be monitored for a road health condition.

The road following system is tested at Ferdowsi University Campus Roads and Regions. For Road following five roads had been selected and tested. After the drone is set on the center of the road, take off would initiate with certain altitude which is based on the user preference. Then, the drone can localize itself based on the maneuvering commands to the center of the road while it follows it with efficient velocity. The altitude in this study varied from 15 m to 20 m, until this threshold won't affect the segmentation module. Also, after several tests, the efficient velocity for the drone was chosen to be 3 $^{m}/_{s}$. Road following system has done well in the experiments without the drone being deviated. Despite that, the drone camera was able to have a good view of the road almost center itself. Also, it's noteworthy that the calibration of the drone helps in more accurate drone command execution.

4.8. Limitations and generalization of the experiment

Considering the whole system has been tested infield, the approach suffers from a couple of limitations. On road detection technique, due to the use of edge detection technique that is implemented. Strong edges (objects in the road i.e., cars or shadows) on the road area cause inconsistency in finding the road boundaries or even miscalculating in finding the vanishing point. Though these limitations were rare on the tested dataset as shown in Fig. 9. Although, the proposed methodology shows promising result in these experiments. It has only been tested in limited number of flights in a metropolitan area. Hence, it's advised for future works to expand the methodology and experiments further to adapt the platform methodology on wide range of roads, e.g., rural, mountain, and dark road areas.

In addition, there are also several limitations in controlling the UAV. Due to the restriction that it's in the UAV manufacturer SDK, controlling the UAV to its best is impossible. Hence, the system was able to control the drone only by some basic maneuvering commanding.

Also, to be noted in our study the accuracy of the GPS coordinates which it's been used for localizing the resulted segments was around ±0.5 m. This accuracy although it's not perfect but shows promising result for road monitoring task.

4.9. Experimental results

After the whole system is processed through, road segments with spatial and temporal information are produced. Fig. 10 Shows some examples of the results that is been produced infield. As its illustrated in Fig. 10, whenever a successful segmentation happens with utilizing the UAV SDK, here DJI SDK, the temporal and spatial information will be assigned to each segment for further investigation capability.

5. Conclusion

Automating road monitoring tasks is an effective improvement for road maintenance planning. UAVs are a better asset in automating road monitoring, as compared with other automatic inspection vehicles. In this study, a fully automatic system is introduced that automates road monitoring with an out-of-shelf UAV. The UAV detects and segmentizes the road as it is following the road. The system showed it can work in real-time upon tests and its efficiency. The road detection algorithm was based on a RANSAC vanishing point and geometry features of the road in the referencing point. And centroid tracking-based algorithm was used for the segmentation road uniquely. Then, a centerline following-based algorithm was used for the road following system. After segmentation, the resulted segment is added with its geolocation tag that was gotten from IMU, to be helpful in monitoring later on and the whole procedure has been tested infield. For future works, we will consider applying deep learning methods such as CNN to increase the efficiency of the system in general on the dataset. Furthermore, to decrease complexity in the proposed approach we will design an end-to-end system for autopilot drones in asphalt surface monitoring instead of using a system with three separate main parts.

Authorship contributions

Conception and design of study: **Hormazd Ranjbar, Perry Forsythe, Alireza Ahmadian Fard Fini, Mojtaba Maghrebi, Travis S. Waller,** acquisition of data **Hormazd Ranjbar, Mojtaba Maghrebi,** analysis and/or interpretation of data: **Hormazd Ranjbar, Alireza Ahmadian Fard Fini, Mojtaba Maghrebi,** Drafting the manuscript **Hormazd Ranjbar, Alireza Ahmadian Fard Fini, Mojtaba Maghrebi,** revising the manuscript critically for important intellectual content: **Hormazd Ranjbar, Alireza Ahmadian Fard Fini, Mojtaba Maghrebi,** Approval of the version of the manuscript to be published (the names of all authors must be listed)

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

All persons who have made substantial contributions to the work reported in the manuscript (e.g., technical help, writing and editing assistance, general support), but who do not meet the criteria for authorship, are named in the Acknowledgements and have given us their written permission to be named. If we have not included an Acknowledgements, then that indicates that we have not received substantial contributions from non-authors.

References

[1] S.D. Indicators, Road Infrastructure and Economic Development, 1992.
[2] M. Huang, et al., LCA and LCCA based multi-objective optimization of pavement maintenance, J. Clean. Prod. 283 (2021), 124583.
[3] J. Zhu, T. Ma, Z. Fang, Characterization of agglomeration of reclaimed asphalt pavement for cold recycling, Construct. Build. Mater. 240 (2020), 117912.
[4] S. Burningham, N. Stankevich, Why Road Maintenance Is Important and How to Get it Done, 2005.
[5] J. Eriksson, et al., The pothole patrol: using a mobile sensor network for road surface monitoring, in: Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services, 2008.
[6] K.-I. Park, Road Monitoring Method for a Vehicle and a System Thereof, Google Patents, 2005.
[7] P.M. Asaro, The labor of surveillance and bureaucratized killing: new subjectivities of military drone operators, Soc. Semiotic. 23 (2) (2013) 196–224.
[8] M.A. Ma'sum, et al., Simulation of intelligent unmanned aerial vehicle (UAV) for military surveillance, in: International Conference on Advanced Computer Science and Information Systems, ICACSIS), 2013, 2013.
[9] C.C. Murray, A.G. Chu, The flying sidekick traveling salesman problem: optimization of drone-assisted parcel delivery, Transport. Res. C Emerg. Technol. 54 (2015) 86–109.
[10] A. Goodchild, J. Toy, Delivery by drone: an evaluation of unmanned aerial vehicle technology in reducing CO2 emissions in the delivery service industry, Transport. Res. Transport Environ. 61 (2018) 58–67.
[11] C. Zhang, J.M. Kovacs, The application of small unmanned aerial systems for precision agriculture: a review, Precis. Agric. 13 (6) (2012) 693–712.
[12] V. Puri, A. Nayyar, L. Raja, Agriculture drones: a modern breakthrough in precision agriculture, J. Stat. Manag. Syst. 20 (4) (2017) 507–518.
[13] J. Irizarry, M. Gheisari, B.N. Walker, Usability assessment of drone technology as safety inspection tools, J. Inf. Technol. Construct. 17 (12) (2012) 194–212.
[14] M. Quaritsch, et al., Networked UAVs as aerial sensor network for disaster management applications, E I Elektrotechnik Inf. 127 (3) (2010) 56–63.
[15] S. Chowdhury, et al., Drones for disaster response and relief operations: a continuous approximation model, Int. J. Prod. Econ. 188 (2017) 167–184.
[16] B. Rabta, C. Wankmüller, G. Reiner, A drone fleet model for last-mile distribution in disaster relief operations, Int. J. Disaster Risk Reduc. 28 (2018) 107–112.

[17] J.C. Hodgson, et al., Precision wildlife monitoring using unmanned aerial vehicles, Sci. Rep. 6 (1) (2016), 22574.
[18] S. Lee, Y. Choi, Reviews of unmanned aerial vehicle (drone) technology trends and its applications in the mining industry, Geosystem Eng. 19 (4) (2016) 197–204.
[19] Y. Pan, et al., Detection of asphalt pavement potholes and cracks based on the unmanned aerial vehicle multispectral imagery, IEEE J. Sel. Top. Appl. Earth Obs. Rem. Sens. 11 (10) (2018) 3701–3712.
[20] A. Puri, A Survey of Unmanned Aerial Vehicles (UAV) for Traffic Surveillance, Department of computer science and engineering, University of South Florida, 2005, pp. 1–29.
[21] F. Nex, F. Remondino, UAV for 3D mapping applications: a review, Applied geomatics 6 (1) (2014) 1–15.
[22] H. Kong, J.-Y. Audibert, J. Ponce, General road detection from a single image, IEEE Trans. Image Process. 19 (8) (2010) 2211–2220.
[23] T. Kühnl, F. Kummert, J. Fritsch, Monocular road segmentation using slow feature analysis, in: IEEE Intelligent Vehicles Symposium (IV), IEEE, 2011, 2011.
[24] J.M. Alvarez, et al., Road scene segmentation from a single image, in: European Conference on Computer Vision, Springer, 2012.
[25] A. Laddha, et al., Map-supervised road detection, in: IEEE Intelligent Vehicles Symposium (IV), IEEE, 2016, 2016.
[26] B. Wang, V. Frémont, S.A. Rodríguez, Color-based road detection and its evaluation on the KITTI road benchmark, in: IEEE Intelligent Vehicles Symposium Proceedings, IEEE, 2014, 2014.
[27] Z. Chen, J. Zhang, D. Tao, Progressive lidar adaptation for road detection, IEEE/CAA Journal of Automatica Sinica 6 (3) (2019) 693–702.
[28] A.M. Kumar, P. Simon, Review of lane detection and tracking algorithms in advanced driver assistance system, Int. J. Comput. Sci. Inf. Technol. 7 (4) (2015) 65–78.
[29] M. Mokhtarzade, M.V. Zoej, Road detection from high-resolution satellite images using artificial neural networks, Int. J. Appl. Earth Obs. Geoinf. 9 (1) (2007) 32–40.
[30] H. Zhou, et al., Efficient road detection and tracking for unmanned aerial vehicle, IEEE Trans. Intell. Transport. Syst. 16 (1) (2014) 297–309.
[31] C. Henry, S.M. Azimi, N. Merkle, Road segmentation in SAR satellite images with deep fully convolutional neural networks, Geosci. Rem. Sens. Lett. IEEE 15 (12) (2018) 1867–1871.
[32] V. Mnih, G.E. Hinton, Learning to detect roads in high-resolution aerial images, in: European Conference on Computer Vision, Springer, 2010.
[33] J. Hu, et al., Road network extraction and intersection detection from aerial images by tracking road footprints, IEEE Trans. Geosci. Rem. Sens. 45 (12) (2007) 4144–4157.
[34] X. Huang, L. Zhang, P. Li, Classification and extraction of spatial features in urban areas using high-resolution multispectral imagery, Geosci. Rem. Sens. Lett. IEEE 4 (2) (2007) 260–264.
[35] X. Huang, L. Zhang, Road centreline extraction from high-resolution imagery based on multiscale structural features and support vector machines, Int. J. Rem. Sens. 30 (8) (2009) 1977–1987.
[36] Y. He, H. Wang, B. Zhang, Color-based road detection in urban traffic scenes, IEEE Trans. Intell. Transport. Syst. 5 (4) (2004) 309–318.
[37] K.-Y. Chiu, S.-F. Lin, Lane detection using color-based segmentation, in: IEEE Proceedings. Intelligent Vehicles Symposium, IEEE, 2005, 2005.
[38] T.-Y. Sun, S.-J. Tsai, V. Chan, HSI color model based lane-marking detection, in: IEEE Intelligent Transportation Systems Conference, IEEE, 2006, 2006.
[39] B. Southall, C.J. Taylor, Stochastic road shape estimation, in: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV, IEEE, 2001, 2001.
[40] B. Yu, A.K. Jain, Lane boundary detection using a multiresolution hough transform, in: Proceedings of International Conference on Image Processing, IEEE, 1997.
[41] A. Kaske, R. Husson, D. Wolf, Chi-square fitting of deformable templates for lane boundary detection, in: IAR Annual Meeting, 1995.
[42] C.R. Jung, C.R. Kelber, A robust linear-parabolic model for lane following, in: Proceedings. 17th Brazilian Symposium on Computer Graphics and Image Processing, IEEE, 2004.
[43] Y. Wang, E.K. Teoh, D. Shen, Lane detection and tracking using B-Snake, Image Vis Comput. 22 (4) (2004) 269–280.
[44] Q. Wang, J. Gao, Y. Yuan, Embedding structured contour and location prior in siamesed fully convolutional networks for road detection, IEEE Trans. Intell. Transport. Syst. 19 (1) (2017) 230–241.
[45] L. Caltagirone, et al., Fast LIDAR-based road detection using fully convolutional neural networks, in: 2017 Ieee Intelligent Vehicles Symposium (Iv), IEEE, 2017.
[46] G.L. Oliveira, W. Burgard, T. Brox, Efficient deep models for monocular road segmentation, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2016.
[47] G. Dong, et al., Real-time high-performance semantic image segmentation of urban street scenes, IEEE Trans. Intell. Transport. Syst. 22 (6) (2020) 3258–3274.
[48] J. Sparbert, K. Dietmayer, D. Streller, Lane detection and street type classification using laser range images, in: ITSC 2001. 2001 IEEE Intelligent Transportation Systems, IEEE, 2001. Proceedings (Cat. No. 01TH8585).
[49] B. Ma, S. Lakshmanan, A.O. Hero, Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion, IEEE Trans. Intell. Transport. Syst. 1 (3) (2000) 135–147.
[50] M. Bertozzi, A. Broggi, GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection, IEEE Trans. Image Process. 7 (1) (1998) 62–81.
[51] W.T. Freeman, E.H. Adelson, The design and use of steerable filters, IEEE Trans. Pattern Anal. Mach. Intell. 13 (9) (1991) 891–906.
[52] J.C. McCall, M.M. Trivedi, Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation, IEEE Trans. Intell. Transport. Syst. 7 (1) (2006) 20–37.
[53] Y. Alon, A. Ferencz, A. Shashua, Off-road path following using region classification and geometric projection constraints, in: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), IEEE, 2006.
[54] S. Rathinam, et al., Autonomous searching and tracking of a river using an UAV, in: 2007 American Control Conference, Ieee, 2007.
[55] S. Rathinam, Z.W. Kim, R. Sengupta, Vision-based monitoring of locally linear structures using an unmanned aerial vehicle, J. Infrastruct. Syst. 14 (1) (2008) 52–63.
[56] E. Frew, et al., Vision-based road-following using a small autonomous aircraft, in: 2004 IEEE Aerospace Conference Proceedings, IEEE, 2004. IEEE Cat. No. 04TH8720).
[57] K. Chaudhury, S. DiVerdi, S. Ioffe, Auto-rectification of user photos, in: IEEE International Conference on Image Processing (ICIP), IEEE, 2014, 2014.
[58] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell. 8 (6) (1986) 679–698.
[59] C. Galamhos, J. Matas, J. Kittler, Progressive probabilistic Hough transform for line detection, in: Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), IEEE, 1999.
[60] M.A. Fischler, R.C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.
[61] S. Suzuki, Topological structural analysis of digitized binary images by border following, Comput. Vis. Graph Image Process 30 (1) (1985) 32–46.
[62] DJI mobile SDK, Available from: https://developer.dji.com/mobile-sdk/.