

Pothole Detection Based on Disparity Transformation and Road Surface Modeling

Rui Fan¹, Member, IEEE, Umar Ozgunalp, Member, IEEE, Brett Hosking, Member, IEEE, Ming Liu², Senior Member, IEEE, and Ioannis Pitas, Fellow, IEEE

Abstract—Pothole detection is one of the most important tasks for road maintenance. Computer vision approaches are generally based on either 2D road image analysis or 3D road surface modeling. However, these two categories are always used independently. Furthermore, the pothole detection accuracy is still far from satisfactory. Therefore, in this paper, we present a robust pothole detection algorithm that is both accurate and computationally efficient. A dense disparity map is first transformed to better distinguish between damaged and undamaged road areas. To achieve greater disparity transformation efficiency, golden section search and dynamic programming are utilized to estimate the transformation parameters. Otsu's thresholding method is then used to extract potential undamaged road areas from the transformed disparity map. The disparities in the extracted areas are modeled by a quadratic surface using least squares fitting. To improve disparity map modeling robustness, the surface normal is also integrated into the surface modeling process. Furthermore, random sample consensus is utilized to reduce the effects caused by outliers. By comparing the difference between the actual and modeled disparity maps, the potholes can be detected accurately. Finally, the point clouds of the detected potholes are extracted from the reconstructed 3D road surface. The experimental results show that the successful detection accuracy of the proposed system is around 98.7% and the overall pixel-level accuracy is approximately 99.6%.

Index Terms—Pothole detection, computer vision, road surface modeling, disparity map, golden section search, dynamic programming, surface normal.

I. INTRODUCTION

ROAD potholes are considerably large structural failures on the road surface. They are caused by contraction and

expansion of the road surface as rainwater permeates into the ground [1]. To ensure traffic safety, it is crucial and necessary to frequently inspect and repair road potholes [2]. Currently, potholes are regularly detected and reported by certified inspectors and structural engineers [3]. This task is, however, time-consuming and tedious [4]. Furthermore, the detection results are always subjective, because they depend entirely on personnel experience [5]. Therefore, automated pothole detection systems have been developed to recognize and localize potholes both efficiently and objectively.

Over the past decade, various technologies, such as active and passive sensing, have been utilized to acquire road data and aid personnel in detecting road potholes [5]. For example, Tsai and Chatterjee [6] mounted two laser scanners on a digital inspection vehicle (DIV) to collect 3D road surface data. These data were then processed using either semi or fully automatic methods for pothole detection. Such systems ensure personnel safety, but also reduce the need for manual intervention [6]. Furthermore, by comparing the road data collected over different periods, the traffic flow can be evaluated and the future road condition can be predicted [2]. The remainder of this section presents the state-of-the-art pothole detection algorithms and highlights the motivation, contributions and outline of this paper.

A. State of the Art in Road Pothole Detection

1) *2D Image Analysis-Based Pothole Detection Algorithms:* There are typically four main steps used in 2D image analysis-based pothole detection algorithms: a) image preprocessing; b) image segmentation; c) shape extraction; d) object recognition [5]. A color or gray-scale road image is first preprocessed, e.g., using morphological filters [7], to reduce image noise and enhance the pothole outline [5], [8]. The preprocessed road image is then segmented using histogram-based thresholding methods, such as Otsu's [9] or the triangle [10] method. Otsu's method minimizes the intra-class variance and performs better in terms of separating damaged and undamaged road areas [9]. The extracted region is then modeled by an ellipse [10]. Finally, the image texture within the ellipse is compared with the undamaged road area texture. If the former is coarser than the latter, the ellipse is considered to be a pothole [5].

However, both color and gray-scale image segmentation techniques are severely affected by various factors, most notably by poor illumination conditions [11]. Therefore,

Manuscript received February 16, 2019; revised June 14, 2019; accepted July 24, 2019. Date of publication August 22, 2019; date of current version October 9, 2019. This work was supported in part by the National Natural Science Foundation of China, under Grant U1713211, in part by the Research Grant Council of Hong Kong SAR Government, China, under Project 11210017 and Project 21202816, and in part by the Shenzhen Science, Technology and Innovation Commission (SZSTI), under Grant JCYJ20160428154842603, awarded to Prof. Ming Liu. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chia-Kai Liang. (Corresponding author: Ming Liu.)

R. Fan and M. Liu are with the Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology, Hong Kong (e-mail: rui.fan@ieee.org; celium@ust.hk).

U. Ozgunalp is with the Department of Electrical and Electronics Engineering, Cyprus International University, 99258 Lefkosa, Mersin 10, Turkey (e-mail: uozgunalp@ciu.edu.tr).

B. Hosking is with the National Oceanography Centre, Southampton SO14 3ZH, U.K. (e-mail: wilski@noc.ac.uk).

I. Pitas is with the School of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece (e-mail: pitas@aia.csd.auth.gr).

Digital Object Identifier 10.1109/TIP.2019.2933750

some authors proposed to perform segmentation on the depth maps, which has shown to achieve better performance when separating damaged and undamaged road areas [6], [11]. Furthermore, the shapes of actual potholes are always irregular, making the geometric and textural assumptions occasionally unreliable. Moreover, the pothole's 3D spatial structure cannot always be explicitly illustrated in 2D road images [12]. Therefore, 3D road surface information is required to measure pothole volumes. In general, 3D road surface modeling-based pothole detection algorithms are more than capable of overcoming the aforementioned disadvantages.

2) *3D Road Surface Modeling-Based Pothole Detection Algorithms*: The 3D road data used for pothole detection is generally provided by laser scanners [6], Microsoft Kinect sensors [11], or passive sensors [4], [12]–[16]. Laser scanners mounted on DIVs are typically used for accurate 3D road surface reconstruction. However, the purchase of laser scanning equipment and its long-term maintenance are still very expensive [3]. The Microsoft Kinect sensors were initially designed for indoor use. Therefore, they greatly suffer from infra-red saturation in direct sunlight [17]. For this reason, passive sensors, such as a single movable camera or multiple synchronized cameras, are more suitable for acquiring 3D road data and for pothole detection [12], [18]. For example, Zhang and Elaksher [13] mounted a single camera on an unmanned aerial vehicle (UAV) to reconstruct the road surface via Structure from Motion (SfM) [18]. A variety of stereo vision-based pothole detection methods have been developed as well [15], [16]. The 3D point cloud generated from a disparity map was interpolated into a quadratic surface using least squares fitting (LSF) [15]. The potholes were then recognized by comparing the difference between the 3D point cloud and the fitted quadratic surface. In [16], the surface modeling was performed on disparity maps instead of the point clouds, and random sample consensus (RANSAC) [19] was used to improve the pothole detection robustness.

B. Motivation

Currently, laser scanning is still the main technology used to provide 3D road information for pothole detection, while other technologies, such as passive sensing, are under-utilized [2]. However, DIVs are not widely used, primarily because of the initial cost but also because their routine operation and long-term maintenance are still very costly [10]. Therefore, the trend is to equip DIVs with inexpensive, portable and durable sensors, such as digital cameras, for 3D road data acquisition. Stereo road image pairs can be used to calculate the disparity maps [14], which essentially represent the 3D road surface geometry. Recently, due to some major advances in computer stereo vision, road surface geometry can be reconstructed with a three-millimeter accuracy [4], [12]. Additionally, stereo cameras used for road data acquisition are inexpensive, portable and adaptable for different DIV types.

So far, comprehensive studies have been made in both 2D image analysis-based and 3D road surface modeling-based pothole detection. Unfortunately, these algorithms are always used independently [5]. Furthermore, pothole detection accuracy is still far from satisfactory [5]. Exploring effective

approaches for disparity map preprocessing, by applying 2D image processing algorithms, is therefore also a popular area of research that requires more attention. Only the disparities in the potential undamaged road areas are then used for disparity map modeling.

Moreover, the surface normal vector is a very important descriptor, which is, however, rarely utilized in existing 3D road surface modeling-based pothole detection algorithms. In this paper, we improve disparity map modeling by eliminating the disparities whose surface normals differ significantly from the expected ones.

C. Novel Contributions

In this paper, a robust stereo vision-based pothole detection system is introduced. The main contributions are: a) a novel disparity transformation algorithm; b) a robust disparity map modeling algorithm; c) three pothole detection datasets which have been made publicly available for research purposes. These datasets are also used in our experiments for assessing pothole detection accuracy.

Since the disparities in damaged road areas can severely affect the accuracy of disparity modeling, we first transform the disparity maps to better distinguish between damaged and undamaged road areas. To achieve greater processing efficiency, we use golden section search (GSS) [20] and dynamic programming (DP) [21] to estimate the transformation parameters. Otsu's thresholding method [22] is then performed on the transformed disparity map to extract the undamaged road areas, where the disparities can be modeled by a quadratic surface using LSF. To improve the robustness of disparity map modeling, the surface normal information is also integrated into the modeling process. Furthermore, RANSAC is utilized to reduce the effects caused by any potential outliers. By comparing the difference between the actual and modeled disparity maps, the potholes can be detected effectively. Finally, different potholes are labeled using connected component labeling (CCL) [23] and their point clouds are extracted from the reconstructed 3D road surface.

D. Paper Outline

The remainder of this paper is structured as follows: Section II details the proposed pothole detection algorithm. The experimental results for performance evaluation are illustrated in Section III. Finally, Section IV summarizes the paper and provides recommendations for future work.

II. POTHOLE DETECTION ALGORITHM

The block diagram of the proposed pothole detection algorithm is illustrated in Fig. 1, where the algorithm consists of three main components: a) disparity transformation; b) undamaged road area extraction; c) disparity map modeling and pothole detection.

A. Disparity Transformation

The input of this procedure is a dense disparity map having subpixel accuracy. Since the performance of disparity map

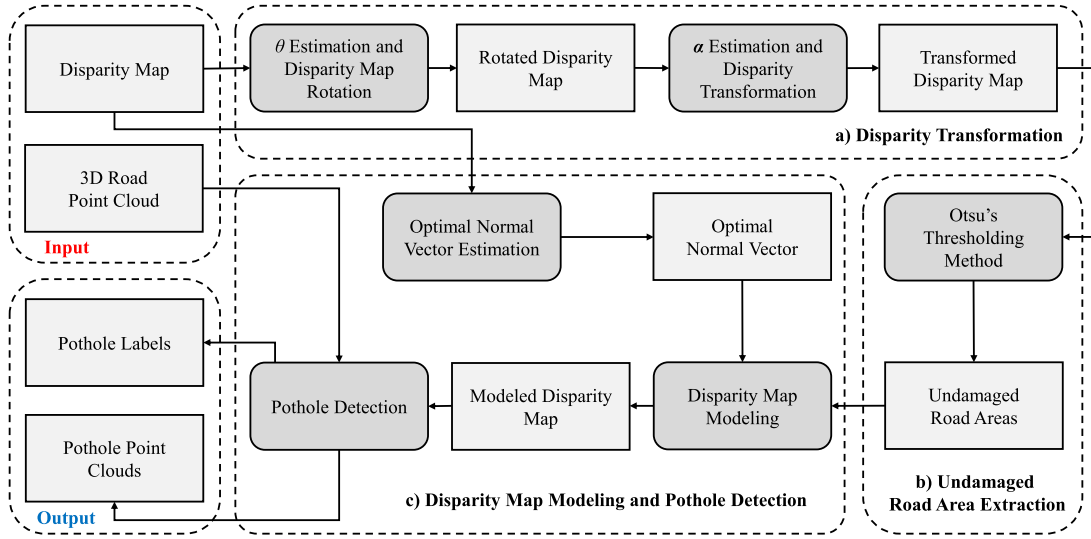


Fig. 1. The block diagram of the proposed pothole detection algorithm.

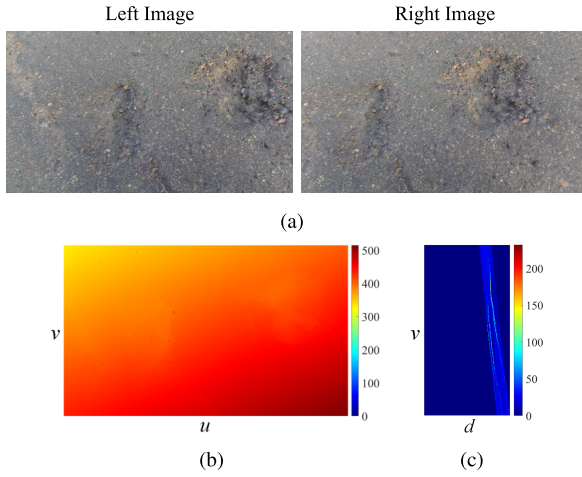


Fig. 2. Disparity map when the roll angle does not equal zero: (a) stereo road images, (b) disparity map, and (c) v-disparity map.

modeling relies entirely on the disparity estimation accuracy, the dense disparity map was obtained from a stereo road image pair (see Fig. 2(a)) through our disparity estimation algorithm [12], where the stereo matching search range propagates iteratively from the bottom of the image to the top, and the subpixel disparity map is refined by iteratively minimizing an energy function with respect to the interpolated correlation parabolas. The disparity map is shown in Fig. 2(b), and its corresponding v-disparity map is shown in Fig. 2(c). A v-disparity map can be created by computing the histogram of each horizontal row of the disparity map. The proposed pothole detection algorithm is based on the work presented in [16], where the disparities of the undamaged road surface are modeled by a quadratic surface as follows:

$$g(u, v) = c_0 + c_1u + c_2v + c_3u^2 + c_4v^2 + c_5uv, \quad (1)$$

where u and v are the horizontal and vertical disparity map coordinates, respectively. The origin of the coordinate system in (1) is at the center of the disparity map. Since in our

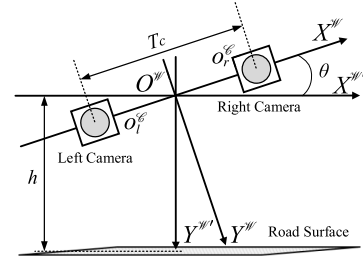


Fig. 3. Roll angle definition in a stereo vision system.

experiments the stereo rig is mounted at a relatively low height, the curvature of the reconstructed road surface is not very high. This makes the values of c_1 , c_3 and c_5 in (1) very close to zero, when the stereo rig is perfectly parallel to the horizontal road surface. In this case, the projection of the road disparities on the v-disparity map can be assumed to be a parabola of the form [21]:

$$g(v) = \alpha_0 + \alpha_1v + \alpha_2v^2. \quad (2)$$

However, in practice, the stereo rig baseline is not always perfectly parallel to the horizontal road surface. This fact can introduce a non-zero roll angle θ (see Fig. 3) into the imaging process, where T_c and h represent the baseline and the height of the stereo rig, respectively. O'_l and O'_r are the origins of the left and right camera coordinate systems, respectively. O'' is the origin of the world coordinate system. An example of the resulting disparity map is shown in Fig. 2(b), where readers can clearly see that the disparity values change gradually in the horizontal direction, which makes the approach of representing the disparity projection using (2) somewhat problematic. In this regard, we first estimate the value of the roll angle. The effects caused by the non-zero roll angle are then eliminated by rotating the disparity map by θ . Finally, the coefficients of the disparity projection model in (2) are estimated, and the disparity map is transformed to better distinguish between damaged and undamaged road areas.

1) θ Estimation and Disparity Map Rotation: Over the past decade, considerable effort has been made to improve the roll angle estimation. The most commonly used device for this task is an inertial measurement unit (IMU). An IMU can measure the angular rate of a vehicle by analyzing the data acquired using different sensors, such as accelerometers, gyroscopes and magnetometers [24], [25]. In these approaches, the road bank angle is always assumed to be zero, and only the roll angle is considered in the estimation process. However, the estimation of both these two angles is always required in many real-world applications. Unfortunately, this cannot be realized with the use of only IMUs [26], [27].

In recent years, many authors have turned their focus towards estimating the roll angle from disparity maps [21], [28]–[31]. For example, the road surface is assumed to be a horizontal ground plane, and an effective roll angle estimation algorithm was proposed based on v-disparity map analysis [28], [29]. In [21], the disparities in a selected small area were modeled by a plane $g(u, v) = c_0 + c_1u + c_2v$. The roll angle was then calculated as $\arctan(-c_1/c_2)$. However, finding a proper disparity map area for plane fitting is always challenging, because the selected area may contain an obstacle or a pothole, which can severely affect the fitting accuracy [30]. Furthermore, the above-mentioned algorithms are only suitable for planar road surfaces. Hence, in this subsection, we introduce a roll angle estimation algorithm, which can work effectively for both planar and non-planar road surfaces.

When the roll angle is equal to zero, the vector $\alpha = [\alpha_0, \alpha_1, \alpha_2]^\top$, storing the disparity projection model coefficients, can be estimated by solving a least squares problem as follows:

$$\alpha = \arg \min_{\alpha} E_0, \quad (3)$$

where

$$E_0 = \mathbf{e}^\top \mathbf{e}, \quad (4)$$

and

$$\mathbf{e} = \mathbf{d} - \mathbf{V}\alpha. \quad (5)$$

The column vector $\mathbf{d} = [d_0, d_1, \dots, d_n]^\top$ stores the disparity values. \mathbf{V} is a matrix of size $(n+1) \times 3$ given as follows:

$$\mathbf{V} = \begin{bmatrix} 1 & v_0 & v_0^2 \\ 1 & v_1 & v_1^2 \\ \vdots & \vdots & \vdots \\ 1 & v_n & v_n^2 \end{bmatrix}. \quad (6)$$

This optimization problem has a closed form solution:

$$\alpha = (\mathbf{V}^\top \mathbf{V})^{-1} \mathbf{V}^\top \mathbf{d}. \quad (7)$$

The minimum energy $E_{0\min}$ can also be obtained by combining (4), (5) and (7):

$$E_{0\min} = \mathbf{d}^\top \mathbf{d} - \mathbf{d}^\top \mathbf{V} (\mathbf{V}^\top \mathbf{V})^{-1} \mathbf{V}^\top \mathbf{d}. \quad (8)$$

However, when the roll angle does not equal zero, the disparity distribution on each row becomes less compact (see Fig. 2(c)).

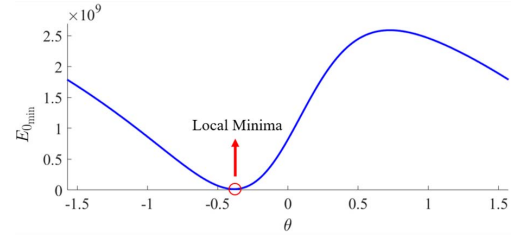


Fig. 4. $E_{0\min}$ function versus θ .

This greatly affects the accuracy of least squares fitting and produces a much higher $E_{0\min}$.

To rotate the disparity map around a given angle θ , each set of original coordinates $[u, v]^\top$ is transformed to a set of new coordinates $[x, y]^\top$ as follows [31]:

$$\begin{aligned} x &= u \cos \theta + v \sin \theta, \\ y &= v \cos \theta - u \sin \theta. \end{aligned} \quad (9)$$

(5) can now be rewritten as follows:

$$\mathbf{e}(\theta) = \mathbf{d} - \mathbf{Y}(\theta)\alpha(\theta), \quad (10)$$

where $\mathbf{Y}(\theta)$ is an $(n+1) \times 3$ matrix:

$$\mathbf{Y}(\theta) = \begin{bmatrix} 1 & y_0(\theta) & y_0(\theta)^2 \\ 1 & y_1(\theta) & y_1(\theta)^2 \\ \vdots & \vdots & \vdots \\ 1 & y_n(\theta) & y_n(\theta)^2 \end{bmatrix}. \quad (11)$$

(7) is rewritten as follows:

$$\alpha(\theta) = (\mathbf{Y}(\theta)^\top \mathbf{Y}(\theta))^{-1} \mathbf{Y}(\theta)^\top \mathbf{d}. \quad (12)$$

(4), (10) and (12) result in the following expression:

$$E_{0\min}(\theta) = \mathbf{d}^\top \mathbf{d} - \mathbf{d}^\top \mathbf{Y}(\theta) (\mathbf{Y}(\theta)^\top \mathbf{Y}(\theta))^{-1} \mathbf{Y}(\theta)^\top \mathbf{d}. \quad (13)$$

Therefore, the main consideration of the proposed roll angle estimation algorithm is to rotate the disparity map at different angles, and find the angle which minimizes $E_{0\min}$. $E_{0\min}$ with respect to different θ is illustrated in Fig. 4. Giving a set of coordinates $[u, v]^\top$, the new coordinate y can be calculated using (9). The corresponding $E_{0\min}$ can be computed from (13). Due to the fact that $\cos(\theta + \pi) = -\cos \theta$ and $\sin(\theta + \pi) = -\sin \theta$, the disparity maps rotated around θ and $\theta + \pi$ are symmetric with respect to the origin of the coordinate system. Namely, (13) outputs the same $E_{0\min}$ no matter how the disparity map is rotated around θ or $\theta + \pi$. Therefore, we set the interval of θ to $(-\pi/2, \pi/2]$. The estimation of θ is achieved by finding the position of the local minima between $-\pi/2$ and $\pi/2$.

However, finding the local minima is a computationally intensive task, because it involves performing the necessary calculations through the whole interval of θ . Furthermore, the step size ε_θ has to be set to a very small and practical value, in order to obtain an accurate value of θ [31]. Hence in this paper, GSS is utilized to reduce the searching times. The procedure of the proposed θ estimation algorithm is given in Algorithm 1, where $\kappa = (\sqrt{5} - 1)/2$ is the golden section ratio [20].

Algorithm 1 θ Estimation Using GSS

Data: disparity map
Result: θ

- 1 set θ_1 and θ_2 to $-\pi/2$ and $\pi/2$, respectively;
- 2 compute $E_{0_{\min}}(\theta_1)$ using Eq. 13;
- 3 compute $E_{0_{\min}}(\theta_2)$ using Eq. 13;
- 4 **while** $\theta_2 - \theta_1 > \varepsilon_\theta$ **do**
- 5 set θ_3 and θ_4 to $\kappa\theta_1 + (1 - \kappa)\theta_2$ and $\kappa\theta_2 + (1 - \kappa)\theta_1$, respectively;
- 6 compute $E_{0_{\min}}(\theta_3)$ using Eq. 13;
- 7 compute $E_{0_{\min}}(\theta_4)$ using Eq. 13;
- 8 **if** $E_{0_{\min}}(\theta_3) > E_{0_{\min}}(\theta_4)$ **then**
- 9 θ_1 is replaced by θ_3 ;
- 10 **else**
- 11 θ_2 is replaced by θ_4 ;
- 12 **end**
- 13 **end**

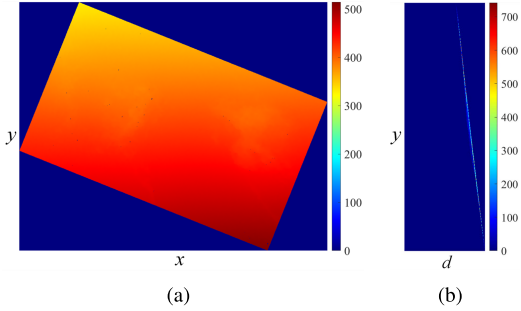


Fig. 5. Elimination of the effects caused by the non-zero roll angle: (a) rotated disparity map and (b) the y-disparity map of Fig. 5(a).

The disparity map is then rotated around θ , as illustrated in Fig. 5(a). A y-disparity map (see Fig. 5(b)) can be created by computing the histogram of each horizontal row of the rotated disparity map. We can observe that the disparity values on each row become more uniform. The evaluation of the proposed roll angle estimation algorithm will be discussed in Section III-B.

2) α Estimation and Disparity Transformation: In this subsection, we utilize DP [21] to extract the road disparity projection model from the y-disparity map. For the purpose of convenient notation, the projection model is also referred to as the target path. The energy of every possible solution is first computed as follows:

$$E_1(d, y) = -m(d, y) + \min_{\tau} [E_1(d + 1, y + \tau) + \lambda\tau] \quad \text{s.t. } \tau \in [\tau_{\min}, \tau_{\max}], \quad (14)$$

where $m(d, y)$ represents the y-disparity value at the position $[d, y]^T$ and λ is a positive smoothness term [21]. E_1 represents the energy of a possible target path in the y-disparity map. τ_{\min} is typically set to 0. τ_{\max} depends entirely on α , and it is set to 10 in this paper. The target path $M = \{(d_i, y_i), i = 0, 1, \dots, n\}$ can be found by minimizing the energy function in (14), where (d_i, y_i) stores the horizontal and vertical coordinates of the target path, respectively.

By substituting the horizontal and vertical coordinates of the target path into (3), (4), (10) and (11), we can obtain α .

Algorithm 2 α Estimation Using DP

Input : M
Output: α

- 1 create a $t \times (s + 3)$ matrix T ;
- 2 **for** $i \leftarrow 1$ **to** t **do**
- 3 randomly select p pairs of coordinates from M ;
- 4 estimate α using Eq. 3;
- 5 $T(i, s + 1 : s + 3) \leftarrow \alpha^T$;
- 6 **for** $j \leftarrow 1$ **to** s **do**
- 7 set the tolerance to $\varepsilon_\alpha / 2^{j-1}$;
- 8 compute the number of inliers n_{inlier} ;
- 9 compute the number of outliers n_{outlier} ;
- 10 compute the ratio $\eta = n_{\text{inlier}} / n_{\text{outlier}}$;
- 11 $T(i, j) \leftarrow \eta$;
- 12 **end**
- 13 **end**
- 14 $j = 0$;
- 15 **do**
- 16 $j \leftarrow j + 1$;
- 17 find the highest η in the j th column of T ;
- 18 **while** the highest η in the j th column of T corresponds to more than one α ;
- 19 find α which corresponds to the highest η in the j th column of T ;

The disparity map can, therefore, be transformed using θ and α . However, the outliers in the target path may greatly affect the accuracy of α estimation. We, therefore, use RANSAC to update the values in α . The full list of procedures involved for α estimation are detailed in Algorithm 2.

RANSAC is iterated t times. Selecting a higher t raises the possibility of finding the best α but also increases the processing time. In order to minimize the trade-off between speed and robustness, t is set to 50 in this paper. In each iteration, we select p pairs of coordinates $[d, y]^T$ from the target path to estimate α . For a smaller p , there is less chance that any outliers will influence optimization. In this paper p is set to 3, which is the smallest possible value for determining α . The ratio η of inliers versus outliers can then be computed with respect to a given tolerance ε_α . The best α corresponds to the highest ratio η . However, the selection of an appropriate ε_α can be challenging, as it is possible that there could be more than one satisfying value for α . Hence, in this paper, the value of ε_α also changes s times. In each iteration, the value of ε_α reduces by half and η is computed. The best α can be determined by finding the highest ratio η . In this paper, the values of ε_α and s are both set to 4.

Finally, each disparity is transformed using the following equation:

$$\tilde{d}(u, v, d, \theta) = [1 \ y(u, v, \theta) \ y(u, v, \theta)^2] \alpha(\theta) - d + \delta, \quad (15)$$

where δ is a constant value set to guarantee that all the transformed disparity values are non-negative. In this paper, we set δ to 30. The transformed disparity map is shown in Fig. 6(a). It can be clearly seen that the disparity values in the undamaged road areas become more uniform, while they

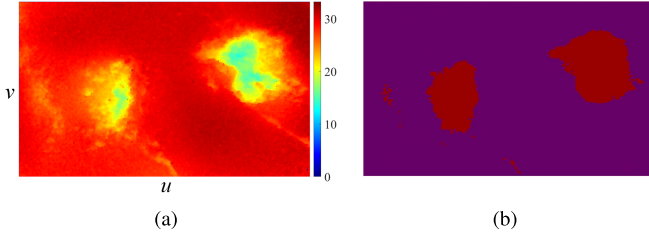


Fig. 6. Disparity transformation and undamaged road area extraction: (a) transformed disparity map and (b) extracted undamaged road areas.

differ significantly from those in the damaged areas (potholes). This makes the extraction of undamaged road regions much simpler.

B. Undamaged Road Area Extraction

Next, we utilize Otsu's thresholding method to segment the transformed disparity map. The segmentation threshold T_0 can be obtained by maximizing the inter-class variance σ_0^2 as follows [22]:

$$\sigma_0^2(T_0) = P_0(T_0)P_1(T_0)[\mu_0(T_0) - \mu_1(T_0)]^2, \quad (16)$$

where

$$P_0(T_0) = \sum_{i=\tilde{d}_{\min}}^{T_0-1} p(i), \quad P_1(T_0) = \sum_{i=T_0}^{\tilde{d}_{\max}} p(i) \quad (17)$$

represent the probabilities of damaged and undamaged road areas, respectively. $p(i)$ is the probability of $\tilde{d} = i$. The average disparity values of the damaged and undamaged road areas are given by:

$$\begin{aligned} \mu_0(T_0) &= \frac{1}{P_0(T_0)} \sum_{i=\tilde{d}_{\min}}^{T_0-1} ip(i), \\ \mu_1(T_0) &= \frac{1}{P_1(T_0)} \sum_{i=T_0}^{\tilde{d}_{\max}} ip(i). \end{aligned} \quad (18)$$

The segmentation result is shown in Fig. 6(b). We can see that the undamaged road area is successfully extracted. However, Otsu's thresholding method will always classify the disparities into two categories, even if the transformed disparity map does not contain any road damage. We, therefore, carry out disparity map modeling to ensure that the potholes are correctly detected.

C. Disparity Map Modeling and Pothole Detection

A common practice in 3D modeling-based pothole detection algorithms [15], [16] is to fit a quadratic surface to either a 3D point cloud or a 2D disparity map. In parallel axis stereo vision, the point cloud is generated from the disparity map as follows [21]:

$$X^{\mathcal{W}} = \frac{uT_c}{d}, \quad Y^{\mathcal{W}} = \frac{vT_c}{d}, \quad Z^{\mathcal{W}} = \frac{fT_c}{d}, \quad (19)$$

where f is the camera focal length. A disparity error larger than one pixel may result in a non-negligible difference in the point cloud [32]. Therefore, disparity map modeling can avoid

such errors generated from (19), producing greater accuracy compared to point cloud modeling.

To model the disparity map, $\mathbf{c} = [c_0, c_1, c_2, c_3, c_4, c_5]^\top$, storing the quadratic surface model coefficients can be estimated as follows:

$$\mathbf{c} = (\mathbf{W}^\top \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{d}, \quad (20)$$

where

$$\mathbf{W} = \begin{bmatrix} 1 & u_0 & v_0 & u_0^2 & v_0^2 & u_0v_0 \\ 1 & u_1 & v_1 & u_1^2 & v_1^2 & u_1v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & u_n & v_n & u_n^2 & v_n^2 & u_nv_n \end{bmatrix}. \quad (21)$$

However, potential outliers can severely affect the accuracy of disparity map modeling and therefore need to be discarded beforehand. In this subsection, a disparity map point is determined as an outlier if it fulfills one of the following conditions:

- it is located in one of the damaged road areas.
- its surface normal vector differs greatly from the optimal one.
- its disparity value differs greatly from the one computed using Eq. 1.

In Section II-B, the undamaged road areas are successfully extracted and we only use the disparities in this area to model the disparity map. The rest of this subsection presents the approaches for determining the outliers which satisfy the last two conditions and the process of modeling the disparity map without using these outliers.

1) *Optimal Normal Vector Estimation*: For each point $\mathbf{p}_i = [u_i, v_i, d_i]^\top$ in the undamaged road area, we would like to estimate a normal vector $\mathbf{n}_i = [n_{ui}, n_{vi}, n_{di}]^\top$ from a set of k points in its neighborhood $\mathbf{Q}_i = [\mathbf{q}_{i1}, \mathbf{q}_{i2}, \dots, \mathbf{q}_{ik}]^\top$, where $\mathbf{q}_{ij} \neq \mathbf{p}_i$. Here, we define the augmented neighbor matrix \mathbf{Q}_i^+ which contains all neighbors and the point \mathbf{p}_i itself as follows:

$$\mathbf{Q}_i^+ = [\mathbf{p}_i, \mathbf{Q}_i]^\top. \quad (22)$$

Existing normal vector estimation methods are generally classified into one of two categories: optimization-based and averaging-based [33]. Although the performance of normal vector estimation depends primarily on the application itself, *PlanePCA* [34], an optimization-based normal vector estimation method, has superior performance in terms of both speed and accuracy. Hence in this subsection, we utilize *PlanePCA* to estimate the normal vectors of the disparities. \mathbf{n}_i can be estimated as follows:

$$\mathbf{n}_i = \arg \min_{\mathbf{n}_i} \left\| \left[\mathbf{Q}_i^+ - \bar{\mathbf{Q}}_i^+ \right] \mathbf{n}_i \right\|_2, \quad (23)$$

where

$$\bar{\mathbf{Q}}_i^+ = \mathbf{1}_{k+1} \left(\frac{1}{k+1} \mathbf{Q}_i^{+\top} \mathbf{1}_{k+1} \right)^\top. \quad (24)$$

$\mathbf{1}_m$ represents an $m \times 1$ vector of ones. Due to the fact that the normal vectors are normalized, they can be projected on a sphere, as shown in Fig. 7, where we can clearly see that the projections are distributed in a small area. Therefore, the optimal normal vector $\hat{\mathbf{n}}$ can be determined by finding the position at which the projections distribute most intensively.

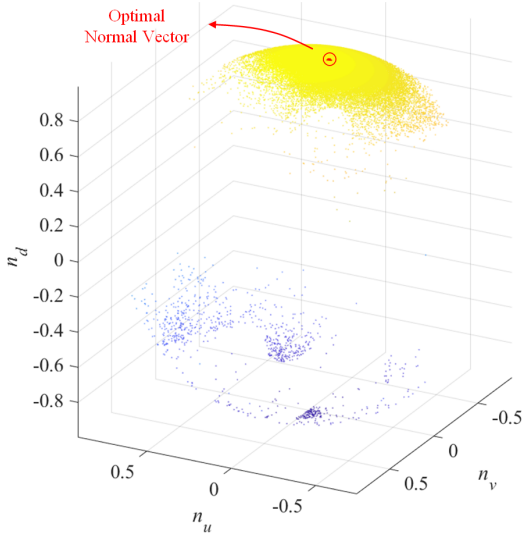


Fig. 7. Surface normal vectors mapping on a sphere.

Since the projection of $\hat{\mathbf{n}}$ is also on the sphere, it can be written in spherical coordinates as follows:

$$\hat{\mathbf{n}} = [\sin \varphi_1 \cos \varphi_2, \sin \varphi_1 \sin \varphi_2, \cos \varphi_1]^\top$$

s.t. $\varphi_1 \in [0, \pi], \varphi_2 \in [0, 2\pi)$. (25)

It can be estimated by minimizing E_2 :

$$E_2 = \mathbf{1}_{n+1}^\top \mathbf{m}, \quad (26)$$

where

$$\mathbf{m} = [-\mathbf{n}_0 \cdot \hat{\mathbf{n}}, -\mathbf{n}_1 \cdot \hat{\mathbf{n}}, \dots, -\mathbf{n}_n \cdot \hat{\mathbf{n}}]^\top. \quad (27)$$

By applying (25) and (27) to (26), the following expressions are derived:

$$\tan \varphi_1 = \frac{\mathbf{1}_{n+1}^\top \mathbf{n}_u \cos \varphi_2 + \mathbf{1}_{n+1}^\top \mathbf{n}_v \sin \varphi_2}{\mathbf{1}_{n+1}^\top \mathbf{n}_d}, \quad (28)$$

$$\tan \varphi_2 = \frac{\mathbf{1}_{n+1}^\top \mathbf{n}_v}{\mathbf{1}_{n+1}^\top \mathbf{n}_u}, \quad (29)$$

where $\mathbf{n}_u = [n_{u0}, n_{u1}, \dots, n_{un}]^\top$, $\mathbf{n}_v = [n_{v0}, n_{v1}, \dots, n_{vn}]^\top$ and $\mathbf{n}_d = [n_{d0}, n_{d1}, \dots, n_{dn}]^\top$. Due to the fact that φ_2 is between 0 and 2π , (29) will have two solutions:

$$\varphi_2 = \arctan \frac{\mathbf{1}_{n+1}^\top \mathbf{n}_v}{\mathbf{1}_{n+1}^\top \mathbf{n}_u} + k\pi, \quad k \in \{0, 1\}. \quad (30)$$

Substituting each φ_2 into (28) produces a value for φ_1 . The two pairs of $[\varphi_1, \varphi_2]^\top$ correspond to the maxima and minima of E_2 , respectively. By substituting each pair of $[\varphi_1, \varphi_2]^\top$ into (26) and comparing the two obtained values, we can find the optimal normal vector. If the angle between $\hat{\mathbf{n}}$ and \mathbf{n}_i exceeds a pre-set threshold ε_n , the corresponding disparity will be considered as an outlier and will not be used for disparity map modeling. In our experiments, we assume that the second category of outliers account for 10% of the undamaged road areas, and therefore, ε_n is set to $\pi/36$ rad. The outliers satisfying the first two conditions can then be successfully removed. The third category of outliers are removed along with the disparity map modeling.

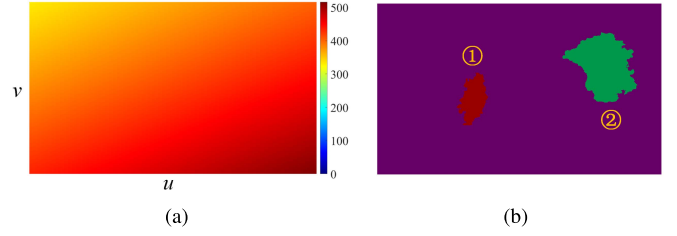


Fig. 8. Disparity map modeling and pothole detection: (a) modeled disparity map and (b) detected potholes.

2) *Disparity Map Modeling*: To model the disparity map with more robustness, we use RANSAC to reduce the effects caused by the third category of outliers described in Section II-C. Here, RANSAC is iterated t times. In each iteration, a subset of disparities are selected randomly to estimate \mathbf{c} . To ensure uniform distribution of the selected disparities, we equally divide the disparity map into a group of square blocks and select one disparity from each block. The disparity block size is $r \times r$. As r becomes smaller, more disparities will be used for surface fitting, which increases the computational complexity. In contrast, selecting a higher value for r results in less computational complexity, but potentially increases noise sensitivity. In this paper, the value of r is set to 125, which produces approximately 100 square blocks for our disparity maps. In each iteration, the differences between the actual and fitted disparities are computed and the ratio η between the inliers and outliers are obtained. \mathbf{c} which corresponds to the highest η is then selected as the desirable surface coefficients. Algorithm 2 presents more details on the least squares fitting using RANSAC. The modeled disparity map is shown in Fig. 8(a).

3) *Pothole Detection*: The potholes can then be detected by finding the regions where the differences between the actual and modeled disparities are larger than a pre-set threshold ε_d . Before labeling different potholes using CCL, the connected components containing fewer than w pixels are removed, because they severely affect the pothole labeling accuracy. Furthermore, the small holes in each connected component are filled, as they are considered to be noise. The selection of ε_d and w is discussed in Section III-D. The detected potholes are shown in Fig. 8(b). Finally, the point clouds of the detected potholes are extracted from the reconstructed 3D road surface. The corresponding results are shown in Fig. 9.

III. EXPERIMENTAL RESULTS

In this section, the performance of the proposed pothole detection algorithm is evaluated both qualitatively and quantitatively. The proposed algorithm was implemented in MATLAB on an Intel Core i7-8700K CPU using a single thread. The following subsections provide details on the experimental set-up and the evaluation of the proposed algorithm.

A. Experimental Set-Up

In this work, we utilized a ZED stereo camera¹ to capture stereo road images. An example of the experimental set-up is

¹<https://www.stereolabs.com/>

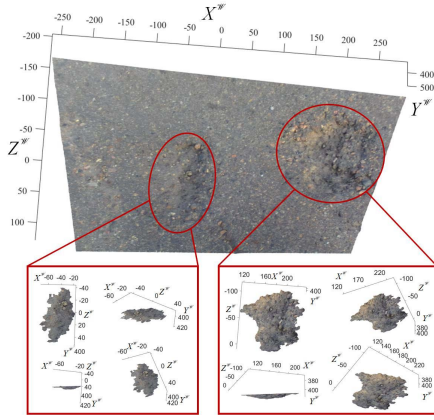


Fig. 9. The point clouds of the detected potholes.



Fig. 10. Experimental set-up for acquiring stereo road images.

shown in Fig. 10. The stereo camera is calibrated manually using the stereo calibration toolbox in MATLAB R2018b. Using the above-mentioned experimental set-up, we created three datasets containing 67 pairs of stereo images. The image resolutions of dataset 1, 2 and 3 are 1028×1730 , 1030×1720 , 1028×1710 pixels, respectively. The disparity maps are estimated using our previously published algorithm [12]. All datasets are publicly available and can be found at: ruirangerfan.com.

The following subsections analyze the accuracy of roll angle estimation, disparity transformation and pothole detection.

B. Evaluation of Roll Angle Estimation

In this subsection, we first analyze the computational complexity of the proposed roll angle estimation algorithm. When estimating the roll angle without using GSS, we have to search through the whole interval of $(-\frac{\pi}{2}, \frac{\pi}{2})$ to find the local minima. Therefore, the computational complexity is $\mathcal{O}(\frac{\pi}{\epsilon_{\theta}})$. In our method, GSS reduces the interval size exponentially. As a result, the interval size then becomes $\kappa^n \pi$ after the n -th iteration. Therefore, the proposed roll angle estimation algorithm reduces the computational complexity to $\mathcal{O}(\log_{\kappa} \frac{\pi}{\epsilon_{\theta}})$. The proposed roll angle estimation algorithm needs 21 iterations to produce a roll angle, with an accuracy higher than $\frac{\pi}{18000}$ rad.

To evaluate the accuracy of the proposed roll angle estimation algorithm, we utilize a synthesized stereo dataset from EISATS [35], [36] where the roll angle is perfectly zero. Some experimental results are shown in Fig. 11. The road areas (see the magenta regions in the first row of Fig. 11) are manually selected and the disparities in these areas are utilized to estimate the roll angle $\hat{\theta}$. The absolute difference between

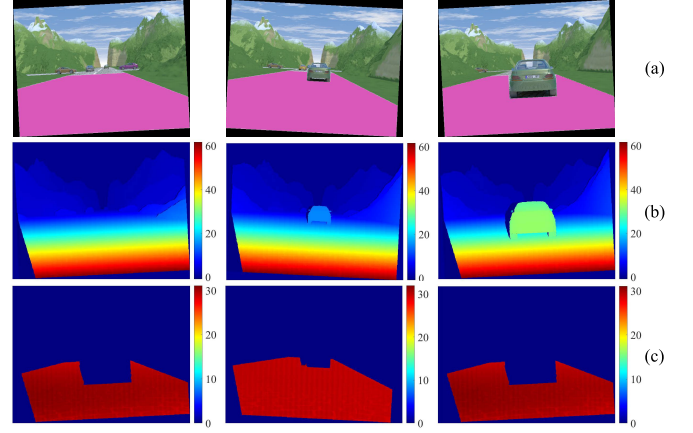


Fig. 11. Experimental results of the EISATS synthesized stereo dataset: (a) left stereo images (the areas in magenta are our manually selected road areas), (b) ground truth disparity maps, and (c) transformed disparity maps.

the actual and estimated roll angles, i.e., $\Delta\theta = |\theta - \hat{\theta}|$, is computed for each frame. The average $\Delta\theta$ is approximately 1.129×10^{-4} rad which is lower than $\frac{\pi}{18000}$ rad. Therefore, the proposed algorithm is capable of estimating the roll angle with high accuracy.

C. Evaluation of Disparity Transformation

Since the datasets we created only contain the ground truth of potholes, KITTI stereo datasets [37], [38] are utilized to quantify the performance of our proposed disparity transformation algorithm (the numbers of disparity maps in the KITTI stereo 2012 and 2015 datasets are 194 and 200, respectively). Some experimental results are shown in Fig. 12. Due to the fact that the proposed algorithm focuses entirely on the road surface, we manually selected a region of interest (see the purple areas in the first row) in each image to evaluate the performance of our algorithm. The corresponding transformed disparity maps are shown in the third row of Fig. 12, where readers can clearly see that the disparities in the road areas tend to have similar values. To quantify the accuracy of the transformed disparity maps, we compute the standard deviation σ_d of the transformed disparity values as follows:

$$\sigma_d = \sqrt{\frac{1}{m+1} \left\| \tilde{\mathbf{d}} - \frac{\tilde{\mathbf{d}}^\top \mathbf{1}_{m+1}}{m+1} \right\|_2^2}. \quad (31)$$

where $\tilde{\mathbf{d}} = [\tilde{d}_0, \tilde{d}_1, \dots, \tilde{d}_m]^\top$ is a column vector storing the transformed disparity values. The average σ_d values of the two KITTI stereo datasets are provided in Table I.

In case 1, the effects caused by the non-zero roll angle are eliminated, while in case 2, the roll angle is assumed to be zero. From Table I, we can observe that the average σ_d values of these two datasets reduce by approximately half when the effects caused by the non-zero roll angle are eliminated. The average σ_d value of these two datasets is only 0.5188 pixels. Therefore, the proposed disparity transformation algorithm performs accurately and the transformed disparity values become very uniform. The runtime of disparity transformation

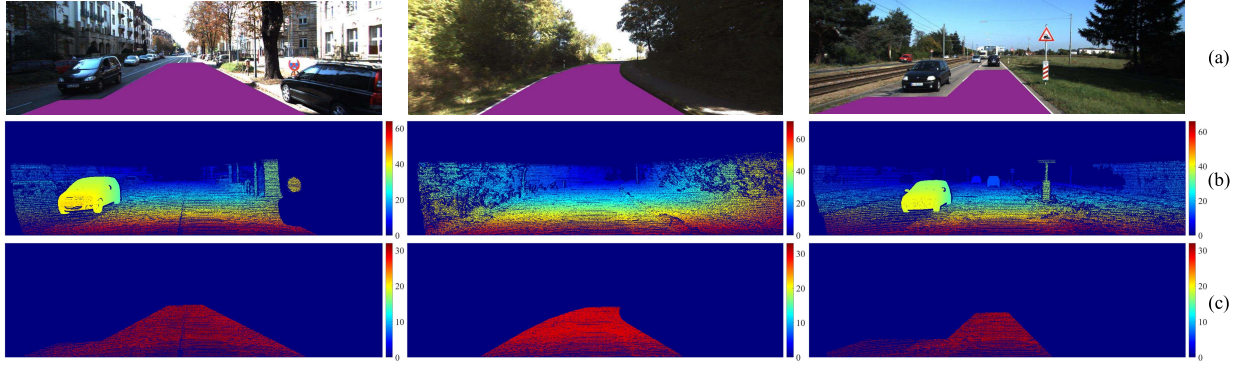


Fig. 12. Experimental results of the KITTI stereo datasets: (a) left stereo images (the areas in purple are the labeled road regions), (b) ground truth disparity maps, and (c) transformed disparity maps.

TABLE I
COMPARISON OF THE AVERAGE σ_d VALUES BETWEEN THE TWO CASES

Dataset	Case 1	Case 2
KITTI stereo 2012 training dataset	0.4862	0.7800
KITTI stereo 2015 training dataset	0.5506	0.9428

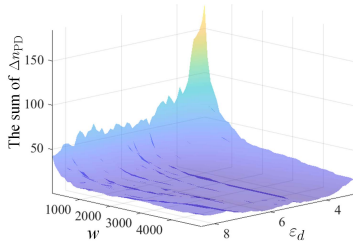


Fig. 13. The sum of Δn_{PD} with respect to different ε_d and w .

is about 142 ms. In the next subsection, we will analyze the accuracy of pothole detection.

D. Evaluation of Pothole Detection

In Section II-C, a set of randomly selected disparities are modeled as a quadratic surface. The potholes are detected by comparing the difference between the actual disparity map and the modeled quadratic surface. If a connected component contains more than w pixels and the disparity difference of each pixel exceeds ε_d , it will be identified as a pothole. In our experiments, we utilize the brute-force search method to find the best values of ε_d and w . The search range for ε_d and w are set to $[3.0, 8.5]$ and $[100, 5000]$, respectively. The step sizes for searching ε_d and w are set to 0.1 and 100, respectively.

For the first step, we go through the whole search range and record the number of detected potholes \hat{n}_{PD} in each frame. The absolute difference Δn_{PD} between each \hat{n}_{PD} and the expected pothole number n_{PD} is then computed. The sum of Δn_{PD} with respect to a pair of given ε_d and w can therefore be obtained, as illustrated in Fig. 13. In our experiments, the least sum of Δn_{PD} is equal to one and it is achieved only when $\varepsilon_d = 6.2$ and $\varepsilon_d = 3100$. The corresponding incorrect pothole detection result is shown in Fig. 14. Incorrect detection occurs when the middle of the first pothole subsides and the selected parameters cause the system to detect two potholes instead of one.

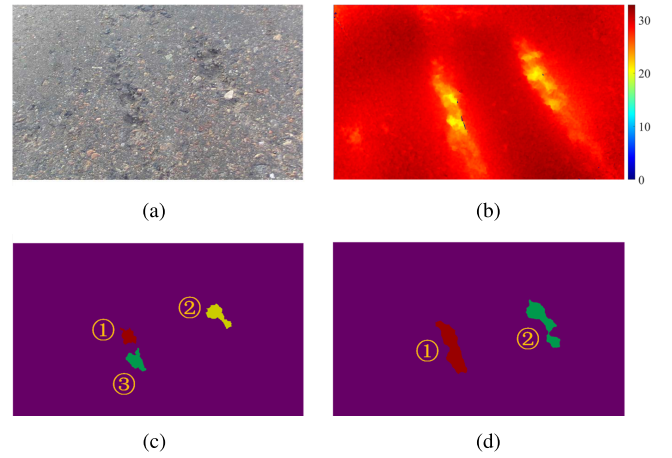


Fig. 14. Experimental result of incorrect pothole detection: (a) left stereo image, (b) transformed disparity map, (c) detection result, and (d) ground truth.

Some examples of successful detection results are shown in the fifth row of Fig. 15, and the corresponding ground truth is shown in the sixth row.

We also compare our proposed algorithm with those produced in [15] and [16]. The pothole detection results obtained using the algorithms presented in [15] and [16] are shown in the third and forth rows of Fig. 15, respectively. The comparative pothole detection results are provided in Table II, where we can see that the successful detection accuracy achieved using [15] and [16] are 73.4% and 84.8%, respectively. Compared to them, our proposed algorithm can detect potholes more accurately with a successful detection accuracy of 98.7%.

We also compare the proposed algorithm with [15] and [16] with respect to the pixel-level precision, recall, F-score and accuracy:

$$\text{precision} = \frac{n_{TP}}{n_{TP} + n_{FP}}, \quad (32)$$

$$\text{recall} = \frac{n_{TP}}{n_{TP} + n_{FN}}, \quad (33)$$

$$\text{F-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}, \quad (34)$$

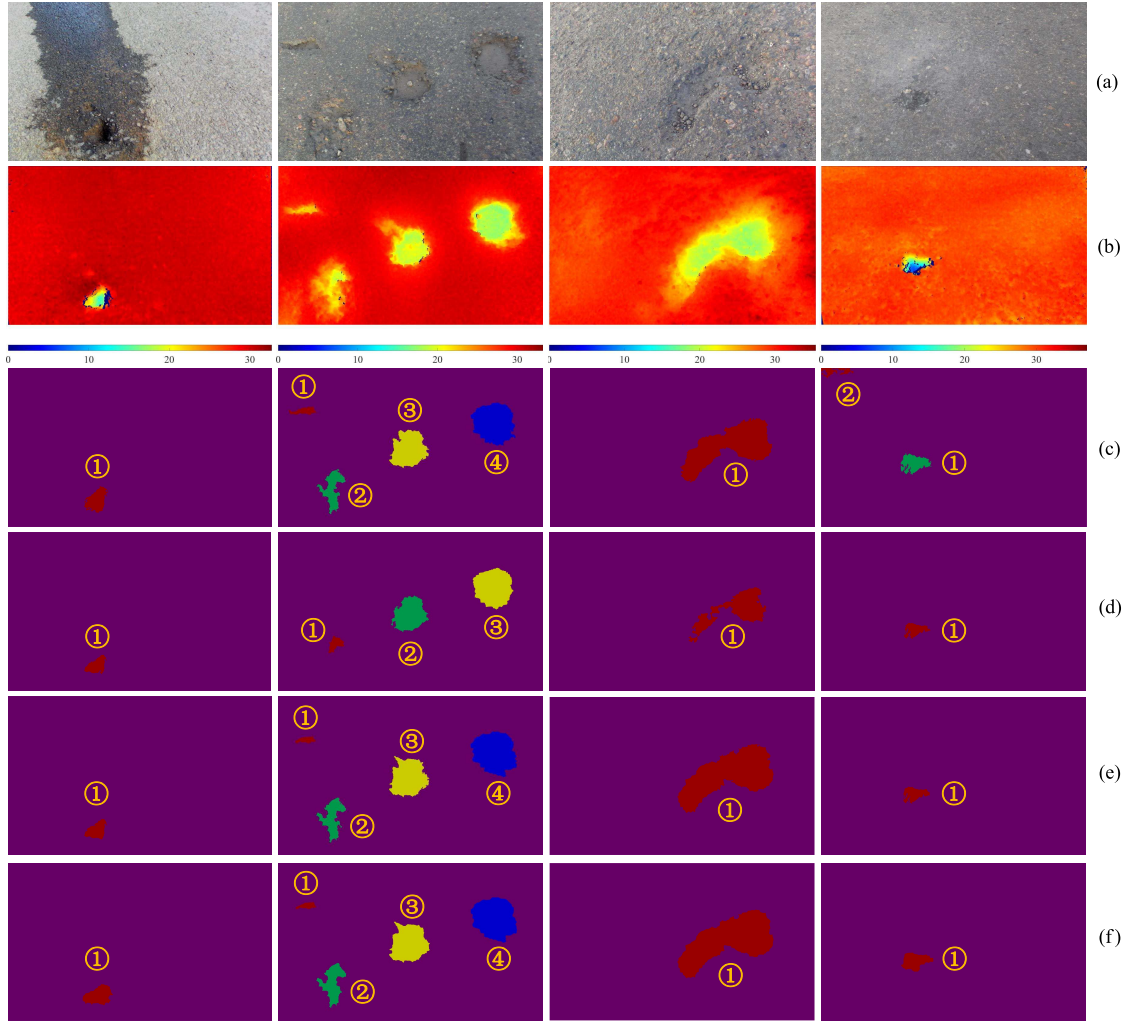


Fig. 15. Experimental results of pothole detection: (a) left stereo images, (b) transformed disparity maps, (c) pothole detection results obtained using the algorithm proposed in [15], (d) pothole detection results obtained using the algorithm presented in [16], (e) pothole detection results obtained using the proposed algorithm, and (f) pothole ground truth.

TABLE II
COMPARISON OF SUCCESSFUL POTHOLE DETECTION ACCURACY

Dataset	Method	Total Potholes	Correct Detection	Incorrect Detection	Misdetection
Dataset 1	algorithm in [15]	22	11	11	0
	algorithm in [16]		22	0	0
	our algorithm		22	0	0
Dataset 2	algorithm in [15]	52	42	10	0
	algorithm in [16]		40	8	4
	our algorithm		51	1	0
Dataset 3	algorithm in [15]	5	5	0	0
	algorithm in [16]		5	0	0
	our algorithm		5	0	0
Total	algorithm in [15]	79	58	21	0
	algorithm in [16]		67	8	4
	our algorithm		78	1	0

$$\text{accuracy} = \frac{n_{TP} + n_{TN}}{n_{TP} + n_{TN} + n_{FP} + n_{FN}}, \quad (35)$$

where n_{TP} , n_{FP} , n_{FN} and n_{TN} are true positive, false positive, false negative and true negative pixel numbers, respectively.

The comparisons with respect to these four indicators are illustrated in Table III. It can be seen that our

proposed algorithm outperforms [15] and [16], in terms of both pixel-level accuracy and F-score. It achieves an intermediate performance in terms of precision and recall. In addition, the precision and recall achieved using our proposed algorithm are very close to the highest values between [15] and [16]. Therefore, the proposed pothole detection algorithm performs both robustly and accurately.

TABLE III
COMPARISON OF PIXEL-LEVEL PRECISION,
RECALL, F-SCORE, AND ACCURACY

Dataset	Method	precision	recall	F-score	accuracy
Dataset 1	[15]	0.5199	0.5427	0.5311	0.9892
	[16]	0.4622	0.9976	0.6317	0.9936
	proposed	0.4990	0.9871	0.6629	0.9940
Dataset 2	[15]	0.9754	0.9712	0.9733	0.9987
	[16]	0.8736	0.9907	0.9285	0.9968
	proposed	0.9804	0.9797	0.9800	0.9991
Dataset 3	[15]	0.6119	0.7714	0.6825	0.9948
	[16]	0.5339	0.9920	0.6942	0.9957
	proposed	0.5819	0.9829	0.7310	0.9961
Total	[15]	0.7799	0.8220	0.8004	0.9942
	[16]	0.6948	0.9921	0.8173	0.9954
	proposed	0.7709	0.9815	0.8635	0.9964

IV. CONCLUSION AND FUTURE WORK

The main contributions of this paper are a novel disparity transformation algorithm and a disparity map modeling algorithm. Using our method, undamaged road areas are better distinguishable in the transformed disparity map and can be easily extracted using Otsu's thresholding method. This greatly improves the robustness of disparity map modeling. To achieve greater processing efficiency, GSS and DP were utilized to estimate the transformation parameters. Furthermore, the disparities, whose normal vectors differ greatly from the optimal one, were also discarded in the process of disparity map modeling, which further improves the accuracy of the modeled disparity map. Finally, the potholes were detected by comparing the difference between the actual and modeled disparity maps. The point clouds of the detected potholes were then extracted from the reconstructed 3D road surface. In addition, we also created three datasets to contribute to stereo vision-based pothole detection research. The experimental results show that the overall successful detection accuracy of our proposed algorithm is around 98.7% and the pixel-level accuracy is approximately 99.6%.

However, the parameters set for pothole detection cannot be applied to all cases. Therefore, we plan to train a deep neural network to detect potholes from the transformed disparity map. Furthermore, road surfaces cannot always be considered to be quadratic. Thus, we aim to design an algorithm to segment the reconstructed road surfaces into a group of localized planes prior to applying the proposed pothole detection algorithm.

REFERENCES

- [1] J. S. Miller and W. Y. Bellinger, "Distress identification manual for the long-term pavement performance program," Office Infrastruct. Res. Develop., Federal Highway Admin., Washington, DC, USA, Tech. Rep. FHWA-HRT-13-092, 2014.
- [2] S. Mathavan, K. Kamal, and M. Rahman, "A review of three-dimensional imaging technologies for pavement distress detection and measurements," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2353–2362, Oct. 2015.
- [3] T. Kim and S. K. Ry, "Review and analysis of pothole detection methods," *J. Emerg. Trends Comput. Inf. Sci.*, vol. 5, no. 8, pp. 603–608, Aug. 2014.
- [4] R. Fan, J. Jiao, J. Pan, H. Huang, S. Shen, and M. Liu, "Real-time dense stereo embedded in a UAV for road inspection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2019. [Online]. Available: http://openaccess.thecvf.com/content_CVPRW_2019/html/UAVision/Fan_Real-Time_Dense_Stereo_Embedded_in_a_UAV_for_Road_Inspection_CVPRW_2019_paper.html
- [5] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Adv. Eng. Inform.*, vol. 29, no. 2, pp. 196–210, 2015.
- [6] Y.-C. Tsai and A. Chatterjee, "Pothole detection and classification using 3D technology and watershed method," *J. Comput. Civil Eng.*, vol. 32, no. 2, Mar. 2018, Art. no. 04017078.
- [7] I. Pitas, *Digital Image Processing Algorithms and Applications*. Hoboken, NJ, USA: Wiley, 2000.
- [8] S. Li, C. Yuan, D. Liu, and H. Cai, "Integrated processing of image and GPR data for automated pothole detection," *J. Comput. Civil Eng.*, vol. 30, no. 6, Nov. 2016, Art. no. 04016015.
- [9] E. Buza, S. Omanovic, and A. Huseinovic, "Pothole detection with image processing and spectral clustering," in *Proc. 2nd Int. Conf. Inf. Technol. Comput. Netw.*, Oct. 2013, p. 4853.
- [10] C. Koch and I. Brilakis, "Pothole detection in asphalt pavement images," *Adv. Eng. Inform.*, vol. 25, no. 3, pp. 507–515, 2011.
- [11] M. R. Jahanshahi, F. Jazizadeh, S. F. Masri, and B. Becerik-Gerber, "Unsupervised approach for autonomous pavement-defect detection and quantification using an inexpensive depth sensor," *J. Comput. Civil Eng.*, vol. 27, no. 6, pp. 743–754, 2013.
- [12] R. Fan, X. Ai, and N. Dahnoun, "Road surface 3D reconstruction based on dense subpixel disparity map estimation," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 3025–3035, Jun. 2018.
- [13] C. Zhang and A. Elaksher, "An unmanned aerial vehicle-based imaging system for 3D measurement of unpaved road surface distresses," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 27, no. 2, pp. 118–129, 2012.
- [14] R. Fan, "Real-time computer stereo vision for automotive applications," Ph.D. dissertation, Dept. Elect. Electron. Eng., Univ. Bristol, Bristol, U.K., 2018.
- [15] Z. Zhang, X. Ai, C. K. Chan, and N. Dahnoun, "An efficient algorithm for pothole detection using stereo vision," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 564–568.
- [16] A. Mikhailuk and N. Dahnoun, "Real-time pothole detection on TMS320c6678 DSP," in *Proc. IEEE Int. Conf. Imaging Syst. Techn. (IST)*, Oct. 2016, pp. 123–128.
- [17] L. Cruz, D. Lucio, and L. Velho, "Kinect and RGBD images: Challenges and applications," in *Proc. 25th SIBGRAPI Conf. Patterns Images Tuts. (SIBGRAPI-T)*, Aug. 2012, pp. 36–49.
- [18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
- [19] M. A. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [20] P. Pedregal, *Introduction to Optimization*, vol. 46. New York, NY, USA: Springer-Verlag, 2006.
- [21] U. Ozgunalp, R. Fan, X. Ai, and N. Dahnoun, "Multiple lane detection algorithm based on novel dense vanishing point estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 621–632, Mar. 2017.
- [22] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
- [23] M. B. Dillencourt, H. Samet, and M. Tamminen, "A general approach to connected-component labeling for arbitrary image representations," *J. ACM*, vol. 39, no. 2, pp. 253–280, Apr. 1992.
- [24] H. E. Tseng, L. Xu, and D. Hrovat, "Estimation of land vehicle roll and pitch angles," *Vehicle Syst. Dyn.*, vol. 45, no. 5, pp. 433–443, May 2007.
- [25] J. Oh and S. B. Choi, "Vehicle roll and pitch angle estimation using a cost-effective six-dimensional inertial measurement unit," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 227, no. 4, pp. 577–590, Apr. 2013.
- [26] J. Ryu, E. J. Rossetter, and J. C. Gerdes, "Vehicle sideslip and roll parameter estimation using GPS," in *Proc. AVEC Int. Symp. Adv. Vehicle Control*, Sep. 2002, pp. 373–380.
- [27] J. Ryu and J. C. Gerdes, "Estimation of vehicle roll and road bank angle," in *Proc. Amer. Control Conf.*, vol. 3, Portland, OR, USA, Jun./Jul. 2005, pp. 2110–2115.
- [28] R. Labayrade and D. Aubert, "A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2003, pp. 31–36.
- [29] P. Skulimowski, M. Owczarek, and P. Strumiłło, "Ground plane detection in 3D scenes for an arbitrary camera roll rotation through 'V-disparity' representation," in *Proc. Federated Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, Sep. 2017, pp. 669–674.

- [30] M. Evans, R. Fan, and N. Dahnoun, "Iterative roll angle estimation from dense disparity map," in *Proc. 7th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2018, pp. 1–4.
- [31] R. Fan, M. J. Bocus, and N. Dahnoun, "A novel disparity transformation algorithm for road segmentation," *Inf. Process. Lett.*, vol. 140, pp. 18–24, Dec. 2018.
- [32] I. Haller and S. Nedevski, "Design of interpolation functions for subpixel-accuracy stereo-vision systems," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 889–898, Feb. 2012.
- [33] K. Klasing, D. Althoff, D. Wollherr, and M. Buss, "Comparison of surface normal estimation methods for range sensing applications," in *Proc. IEEE Int. Conf. Robot. Autom.*, Kobe, Japan, May 2009, pp. 3206–3211.
- [34] C. Wang, H. Tanahashi, H. Hirayu, Y. Niwa, and K. Yamamoto, "Comparison of local plane fitting methods for range data," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Dec. 2001, p. 1.
- [35] T. Vaudrey, C. Rabe, R. Klette, and J. Milburn, "Differences between stereo and motion behaviour on synthetic and real-world stereo sequences," in *Proc. 23rd Int. Conf. Image Vis. Comput. New Zealand*, Nov. 2008, pp. 1–6.
- [36] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers, "Efficient dense scene flow from sparse or dense stereo data," in *Proc. 10th Eur. Conf. Comput. Vis.*, Marseille, France, Oct. 2008, pp. 739–751.
- [37] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.
- [38] M. Menze, C. Heipke, and A. Geiger, "Joint 3D estimation of vehicles and scene flow," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 2, p. 427, Aug. 2015.



Rui Fan received the B.Sc. degree in control science and engineering from the Harbin Institute of Technology, in 2015, and the Ph.D. degree in electrical and electronic engineering from the University of Bristol, in September 2018. Since July 2018, he has been a Research Associate with the Robotics Institute, the Hong Kong University of Science and Technology. His research interests include computer vision, machine learning, image/signal processing, autonomous driving, and high-performance computing.



Umar Ozgunalp received the B.Sc. degree (Hons.) in electrical and electronic engineering from Eastern Mediterranean University, in 2007, the M.Sc. degree in electronic communications and computer engineering from the University of Nottingham, U.K., in 2009, and the Ph.D. degree from the University of Bristol, U.K., in 2016. He is currently an Assistant Professor with the Department of Electrical and Electronics Engineering, Cyprus International University. His research interests include computer vision, pattern recognition, and intelligent vehicles.



signal processing, video coding and compression, computer vision, and machine learning.

Brett Hosking received the M.Sc. degree in electronic and electrical engineering from Loughborough University, U.K., in 2011, and the Ph.D. degree in electronic and electrical engineering from the University of Bristol, U.K., in 2016, under the supervision of Dr. D. Agrafiotis and Prof. D. Bull. In 2017, he joined the National Oceanography Centre, U.K. He has primarily been involved in the research and development of object detection and classification tools in underwater imagery using autonomous underwater vehicles. His research interests include



Ming Liu received the B.A. degree in automation from Tongji University in 2005 and the Ph.D. degree from the Department of Mechanical and Process Engineering, ETH Zurich, in 2013, under the supervision of Prof. R. Siegwart. He is currently an Assistant Professor with the ECE and CSE Department, the Hong Kong University of Science and Technology. He is the Principal Investigator of over 20 projects, supported by RGC, NSFC, ITC, and SZSTI. His research interests include dynamic environment modeling, deep learning for robotics, 3D mapping, machine learning, and visual control. He was a recipient of the Best RoboCup Paper Award from the IEEE IROS 2013, the Innovation Contest Chunhui Cup Winning Award twice in 2012 and 2013, and the 2018 IEEE IROS Toshio Fukuda Young Professional Award.



Ioannis Pitas received the Diploma and Ph.D. degrees in electrical engineering from the Aristotle University of Thessaloniki, Greece.

He was the Chair of computer science with School of Graduate Studies, University of Thessaloniki. Since 1994, he has been a Professor with the Department of Informatics, Aristotle University of Thessaloniki. He has served as a visiting professor with several universities. He has published over 1090 papers and contributed in 50 books in his areas of interest. He edited or coauthored another 11 books. He has also been a member of the program committee of many scientific conferences and workshops. He has participated in 69 Research and Development projects, primarily supported by the European Union. He is/was a Principal Investigator/Researcher in 41 such projects. He leads the big European H2020 Research and Development Project MULTIDRONE. He has over 28 700 citations to his work and an h-index of over 80 (Google Scholar). His current interests include the areas of autonomous systems, image/video processing, machine learning, computer vision, intelligent digital media, human-centered interfaces, affective computing, 3D imaging, and biomedical imaging.

Prof. Pitas is an IEEE Distinguished Lecturer and an EURASIP Fellow. He is the Chair of the IEEE Autonomous Systems Initiative (ASI). He has served as the general or technical chair for four international conferences. In the past, he has served as an associate editor or coeditor for nine international journals.