Chapter

On the Use of Low-Cost RGB-D Sensors for Autonomous Pothole Detection with Spatial Fuzzy c-Means Segmentation

Yashon Ombado Ouma

Abstract

The automated detection of pavement distress from remote sensing imagery is a promising but challenging task due to the complex structure of pavement surfaces, in addition to the intensity of non-uniformity, and the presence of artifacts and noise. Even though imaging and sensing systems such as high-resolution RGB cameras, stereovision imaging, LiDAR and terrestrial laser scanning can now be combined to collect pavement condition data, the data obtained by these sensors are expensive and require specially equipped vehicles and processing. This hinders the utilization of the potential efficiency and effectiveness of such sensor systems. This chapter presents the potentials of the use of the Kinect v2.0 RGB-D sensor, as a low-cost approach for the efficient and accurate pothole detection on asphalt pavements. By using spatial fuzzy c-means (SFCM) clustering, so as to incorporate the pothole neighborhood spatial information into the membership function for clustering, the RGB data are segmented into pothole and non-pothole objects. The results demonstrate the advantage of complementary processing of low-cost multisensor data, through channeling data streams and linking data processing according to the merits of the individual sensors, for autonomous cost-effective assessment of road-surface conditions using remote sensing technology.

Keywords: Kinect RGB-D sensor, pothole detection, spatial fuzzy-c means clustering (SFCM), sensor calibration

1. Introduction

Presently, two approaches are typically used to monitor the condition of pavements: manual distress surveys and automated condition surveys using specially equipped vehicles. Traditionally, in order to determine the serviceability of road pavements, designated pavement officers perform on-site inspection, either by walk-observe-record or by windshield (drive-by) inspection, so as to aggregate the roughness, rutting and surface distresses [1, 2]. With the advancement of sensor technology, numerous automatic pavement evaluation systems have been proposed to aid in pavement condition inspection during the last two decades [3]. Currently, there exist several off-the-shelf commercial systems, which are being widely used

by some of the road maintenance agencies for detailed pavement distress evaluation and exclusive crack analysis. Among which, the Fugro Roadware's ARAN, CSIRO's RoadCrack and Ramböll OPQ's PAVUE are of the world's leading manufacturers offering an integrated full-fledged pavement evaluation system equipped with Global Positioning System (GPS)/Inertial Measurement Unit (IMU) sensors, Light Detection And Ranging (LiDAR) system, high definition video camera, and special lighting illumination systems [2]. Nonetheless, technology for the monitoring of pavement condition does not appear to have kept pace with other technological improvements over the past several years. Furthermore, these pavement monitoring and evaluation approaches remain rather reactive than proactive in terms of detecting distresses and damage, since they merely record the distress that has already appeared, and most of these methods either require significant personnel time or the use of costly equipment. Thus these systems and techniques can only be used cost-effectively on a periodic and or localized basis, and may not allow for continuous long-term monitoring and deployment at the network level, due limitations in hardware and software development and costs.

For sustainable and cost-effective road infrastructure management, the road agencies charged with the responsibility of road maintenance and repairs should be able to continuously collect road condition data within their network, with the objective of building and implementing pavement information and management systems (PIMS) using non-destructive techniques. However, as already stated above, data collection for a whole network such as an entire city or town is expensive and time consuming, if pursued by traditional surveys. Developments in sensor technology for digital image acquisition and computer technology for image data storage and processing can allow the local agencies to use digital image processing for pavement distress analyses. In order to overcome the cost limitations in pavement data collection, this chapter presents a pervasive and 'smart' nature of the low-cost consumer-grade devices, in the acquisition of roadway condition data. By using such devices, no dedicated and expensive platforms and drivers are needed for automated data collection, and are as such suitable in the long-term in terms of costs, implementation and operations for road condition surveys.

Besides the data acquisition systems, in order to enhance the automation of pavement condition monitoring, there have also been advancements in the data collection techniques (e.g., [4–7]), and automated data processing techniques [8–10]. Because of the irregularities in terms of noise and topographic structure of pavement surfaces, more research is still ongoing on the accurate detection, classification and quantification of cracks and potholes. In addition, the computational costs for automated pavement distress detections are expensive, and better approaches are still necessary in the evaluation of the automated crack measurement systems under the various conditions [11].

The commercially available state-of-the-art systems, which comprise of digital camera and laser-illumination module, and laser road-imaging vehicles costs about $150,000. On the other hand, the pavement-surface profiler laser sensors, which are commonly used for measurement of road rutting-depth or surface-roughness, cost in the range of $130,000–$150,000. Comparatively, mobile pavement imaging techniques and manual inspection approaches respectively costs $88.5/mile and $428.8/mile, and the cost of using multi-sensor hybrid systems can range from $541/mile to $933/mile [2]. For fully automated pavement mapping systems, the cost of the imaging sensors and operations defines the purchase pricing, which averages at approximately $697,152 [12]. This chapter presents an approach for the customization of a low-cost imaging system, Kinect v2.0 sensor, as a prototype for cost-effective pavement imaging, and a data processing pipeline for pothole detection and extraction on asphalt pavements.

2. Measurement principle of the Kinect v2.0 RGB-D sensor

The Kinect v2.0 is the successor of the Xtion Pro Live RGB-D camera, called the Kinect v1.0. The version 2.0 Kinect RGB-D camera consists of a color (RGB) camera, an IR illuminator or projector and IR camera (**Figure 1(a)**). While the RGB camera records color information in high definition (HD), the IR projector emits an infrared laser and the IR camera is the sensor for the infrared laser. The Kinect v2 field in the horizontal is 70.6° and 60° in the vertical as depicted in **Figure 1(c)**. The values in the z-direction (depth values), are calculated using the Time of Flight (ToF) principle [16, 17], as shown in Eq. (1), and the x and y values are determined by using the homogeneous image coordinates u and v, and calculated as in Eqs. (2) and (3) [18]. The RGB and IR images acquired with the Kinect v2.0 partially overlap, because the RGB color camera has a wider horizontal field of view (FOV), and IR camera has a larger vertical FOV [15].

$$z = h = \frac{c \cdot \Delta\varphi}{4\pi f} \tag{1}$$

$$x = \frac{u - C_x}{f_x} \tag{2}$$

$$y = \frac{v - C_x}{f_y} \tag{3}$$

where z is the depth measure in meters, $\Delta\varphi$ is the phase shift, c is the speed of light and f is the modulation frequency; x is the horizontal position, u is the vertical image coordinate, C_x is optical center in the X-direction and f_x is the focal length in the X-direction, and y is the vertical position, v is the horizontal image coordinate, C_y is optical center in the Y-direction and f_y is the focal length in the Y-direction. In **Figure 1(b)**, **P** is the measured point on object surface, E is the IR emitter C is the IR sensor, and h or z is the unknown distance of measured point from sensor origin.

For the Kinect v1.0 RGB-D camera, the IR camera analyses a fixed speckle pattern projected by the IR projector and computes depth values by triangulation. This pattern analysis is referred to as the structured light (SL) approach, whereby a memorized IR pattern stored in the RGB-D camera's computer architecture is
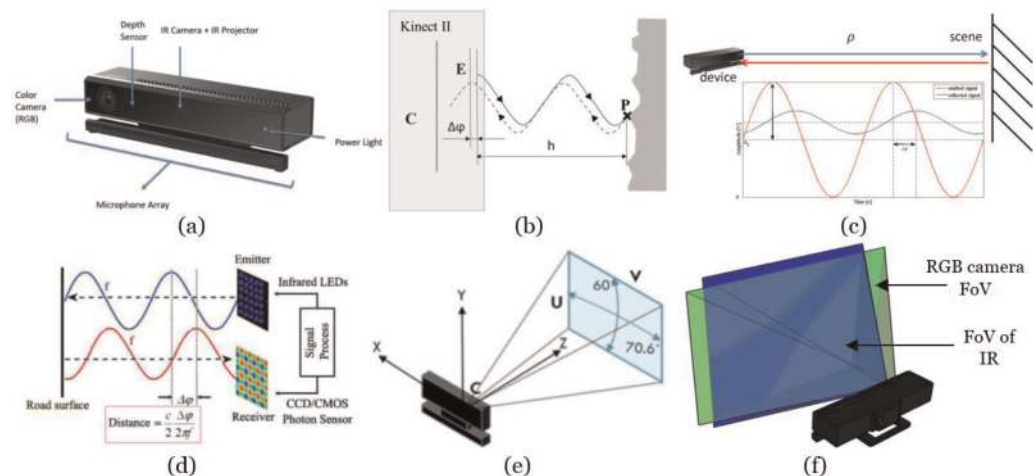

Figure 1.
(a) Kinect sensor v2.0 cameras; (b) and (c) principle of Time of Flight (ToF) phase measurement in Kinect v2.0, and (d) Kinect v2.0 and the field of view geometry [13, 14]. (e) Field of view (FoV) of Kinect v2.0 RGB and IR cameras [15].

projected onto the screen and compared with the current pattern on the screen [19]. If there are any obstacles in the way, the IR pattern changes shape from which the depth values can be deciphered. The Kinect v2.0 however, uses ToF technique to acquire depth values, where the sensor measures the time it takes for the modulated laser pulses from the IR projector to reach the object and then back to the IR camera [13]. The RGB resolution of the Kinect v2.0 is at 1920 × 1080 pixels, and the IR camera has a resolution of 512 × 424 pixels, with corresponding pixel sizes of 3.1 and 10 μm respectively. The collection of the (x,y,z) points results into 3D point

(a)				
Specifications	**Microsoft Kinect v1.0**	**SoftKinetic DS311**	**SoftKinetic DS325**	**SwissRanger SR4000**
Range (short)	N/A	N/A	15 cm–1.5 m	N/A
Range (long)	0.8–4 m	1.5–4.5 m	N/A	0.8–8 m
Resolution (depth)	QVGA (640 × 480)	QVGA (320 × 240)	QVGA (160 × 120)	176 × 144
Field of view (H × V × D)	57.5° × 43.5° × N/A	57.3° × 42° × 73.8°	74° × 58° × 87°	43° × 34° × N/A
Technology (depth sensor)	Light coding	Depth sense	CAPD ToF	Time of Flight (ToF)
Frame rate (depth sensor)	30	25–60	25–60	50
Resolution (RGB)	640 × 480 or 1280 × 960	640 × 480	1280 × 720 (HD)	N/A
Field of view (RGB)	57.3° × 42° × N/A	50° × 40° × 60°	63.2° × 49.3° × 75.2°	N/A
Frame rate (RGB)	30	<25	<25	N/A
Power/data connection	USB 2.0 (1)	USB 2.0 (1)	USB 2.0 (1)	Lumberg M8 Male 3-pin
Size (W × H × D)	27.94 × 7.62 × 7.62 cm	24 × 5.8 × 4 cm	10.5 × 3.1 × 2.7 cm	6.5 × 6.5 × 6.8 cm
Price	$99	$299	$249	$4295

(b)		
Parameter specification	**Kinect v1.0**	**Kinect v2.0**
Resolution of RGB camera (pixel)	640 × 480 or 1280 × 960	1920 × 1080
Resolution of IR and depth camera (pixel)	640 × 480	512 × 424
Field of view (FOV) of color camera	62° × 48.6°	84.1° × 53.8°
Field of view (FOV) of IR and depth image	57.5° × 43.5°	70.6° × 60°
Tilt motor	Yes	No
Maximum skeletal tracking	2	6
Method of depth measurement	Structured light	Time-of-Flight (ToF)
Depth distance working range	0.8–4.0 m	0.5–4.5 m
USB	2.0	3.0
Price	$99	$200

Table 1.
Comparative specifications of Kinect v1.0 and Kinect v2.0 and other low-cost sensors.

cloud. This implies at the acquisition rate of 30 frames per second (fps), every frame of the Kinect v2.0 outputs 217,088 colored 3D points. The advantage that the Kinect v2.0 has over its predecessor Xtion Pro Live (Kinect v1.0), is that since it uses the principle of the ToF instead of relying on projected IR patterns for computing depth, the interference problem is greatly reduced as the sensor does not have to compute distances between neighboring points on the pattern [13]. The other advantage with the Kinect v2.0 over the Xtion, is that the camera has a built in ambient-light rejection method, which makes it possible to use in an outdoor environment with near infrared sources of interference [16]. **Table 1(a)** presents a summary of the differences between the Microsoft Kinect sensor v1.0 and other low-cost sensors, and **Table 1(b)** presents the fundamental characteristics of the Kinect versions 1.0 and 2.0.

3. Low-cost hardware system design and set-up for pavement data acquisition using Kinect v2.0

The establishment and design of an optimal low-cost imaging system, comprising of the hardware platform and peripheral requirements, with interface for Kinect-computer data acquisition, visualization and storage, in both static and dynamic acquisition modes is illustrated in **Figure 2**, and is termed as integrated Mobile Mapping Sensor System (iMMSS). For the implementation of the iMMSS, two main sets of equipment are used: (i) the Kinect v2.0—for RGB, Infrared (IR) and depth data capture, and (ii) a DC-AC power inverter—12 V DC to AC 220 V/200 W output. The power inverter is adaptable to the car charger port for powering the Kinect sensor for static and continuous pavement data acquisition modes. The iMMSS data acquisition system hardware-software set-up is as illustrated in the photo in **Figure 2**. The three main criteria in the field experimentation using the iMMSS comprise of: the shooting angle (vertical and oblique), shooting distance from the pavement, and the overall target positioning. **Figure 2** illustrates the hardware layout and software data capture system. The sensing device is housed within a sensor rack mounted onto the exterior of the wagon. To improve the contrast of the Kinect's laser pattern over the road surfaces, from the reflected IR radiation from sunlight an umbrella was used to block the rays from the sun and to create a shadow.

In terms of data acquisition in static and dynamic mode (**Figure 2**), the Kinect sensor captures depth and color images simultaneously at a frame rate of up to 30 fps. The integration of depth and color data results in a colored point cloud that

Figure 2.
iMMSS hardware-software set-up for road pavement data capture, visualization and storage using the Kinect sensor.

contains about 300,000 points in every frame. By registering the consecutive depth images it is possible to obtain an increased point density, and to create a complete point cloud. To realize the full potential of the sensor for mapping applications an analysis of the systematic and random errors of the data is necessary. The correction of systematic errors is a prerequisite for the alignment of the depth and color data, and relies on the identification of the mathematical model of depth measurement and the calibration parameters involved. The characterization of random errors is important and useful in further processing of the depth data, for example in weighting the point pairs or planes in the registration algorithm [20].

1. Pothole detection and the bias field effect

Under perfect conditions, potholes tend to have two visual properties characterized by: (i) low-intensity areas that are darker than nearby pavement because of road surface irregularity [21], and (ii) the texture inside the potholes being coarser than the nearby pavement [1, 22]. However, as illustrated in [8, 23], the pothole area is not always darker than nearby pavement. Furthermore, the irregularity of the road surface produces shadows at pothole boundaries, which is darker than nearby pavement. These conditions results into the lower accuracy of pothole detection using visual 2D techniques as was reported in [8]. In RGB imagery, pothole detection is influenced by the spill-in and spill-out phenomenon [1, 8], which is typically characterized by the similarities in the defect and non-defect features and regions. These results in the corruption of the defect regions on the pavement, with a smoothly varying intensity inhomogeneity called bias field. Bias is inherent to pavement imaging, and is associated with the imaging equipment limitations and also the pavement surface noise [1, 2].

Bias field in pothole detection can be modeled as a multiplicative component of an observed image, and varies spatially because of inhomogeneities, and can be modeled as in Eq. (4).

$$Y_j = B_j X_j + n \tag{4}$$

where Y_j is the measured image at voxel j; X_j is the true image signal to be restored; B_j is an unknown noise or bias field, and n is the additive zero-mean Gaussian noise. Eq. (4) modeled as an additive component by applying a logarithmic transformation, it is possible to obtain a simplified form as:

$$y_j = x_j + b_j \tag{5}$$

where x_j and y_j are the true and observed log transformed intensities at the jth voxel, respectively, and b_j is the noise or bias field at the jth voxel.

Bias or noise can be corrected by using prospective and retrospective methods. Prospective methods for noise minimization aim at avoiding the intensity inhomogeneities in the image acquisition process. Prospective methods are capable of correcting intensity inhomogeneity induced by the imaging devices; they are not able to remove object-induced effects. Retrospective methods in contrast, rely only on the information in the acquired images, and can thus remove intensity inhomogeneities regardless of their sources. The obvious choice in noise minimization is therefore the retrospective methods, which include filtering, surface fitting, histogram, and segmentation. Among the retrospective methods, segmentation-based approaches are particularly attractive, as they unify the tasks of segmentation and bias correction into a single framework. When an observed pixel y_j is defined as noisy, the neighboring pixels can be used to correct it, since the pixel is expected to

be similar to its surrounding pixel. That is, the data points with similar feature vectors can be grouped into a single cluster and the data points with dissimilar feature vectors are also grouped into different clusters. By using a pre-segmentation clustering algorithm, the Euclidean distance between neighboring pixels is computed and used for the *a priori* clustering. This means that pixels that produce the lowest distance values to their neighbors are categorized as being nearly similar. Two pixels with similar neighboring values are expected to be close to each other, and hence the pixels can be clustered together. On way of minimizing noise through clustering is by using the k-means clustering algorithm, whereby the distance measure between every point $z_j^{(j)}$, and the cluster v_j is optimized by calculating the Euclidean distance measure $\left\| z_i^{(j)} - v_j \right\|^2$. The value of this distance measure function is an indicator of the proximity of the n data points to their cluster prototypes. Once the pre-clustering is carried out, a more robust segmentation approach can then be applied, to cluster the smoothened pavement image.

Image segmentation can be performed using different techniques such as: thresholding, clustering, transform and texture based methods [24]. Histogram-based thresholding is the simplest and often used approach [25]. Many global and local thresholding methods have been developed. While the global thresholds segment the entire image, with a single threshold using the gray-level histogram, the local based thresholds partition the image into a number of sub-images and select a threshold for each of the sub-image. The global thresholding methods select the thresholding based on different criterion such as: Otsu's method [24], minimum error thresholding [26], and entropic method [27]. These one-dimensional (1D) histogram thresholding methods work well when the two consecutive gray levels of the images are distinct. Further, all the 1D thresholding techniques do not combine the spatial information and the gray-level information of the pixels into the segmentation process. The performance of the thresholding techniques will lead to misclassifications in inherently correlated imagery, which are already corrupted by noise and other artifacts.

Real-world images are often ambiguous, with indistinguishable histograms. As such, it is complicated for the classical thresholding techniques to find criterion of similarity or closeness for optimal thresholding. This ambiguity in image segmentation can be solved by using fuzzy set theory, as a probabilistic global image segmentation approach. Using the conventional FCM formulation, each class is assumed to have a uniform value as given by its centroid. Similarly, each data point is also assumed to be independent of every other data point and spatial interaction between data points is not considered. However, for image data, there is strong correlation between neighboring pixels. In addition, due to the intensity non-uniformity artifacts, the data in a class no longer have a uniform value. Thus to realize meaningful segmentation results, the conventional FCM algorithm has to be modified to take into account both local spatial continuity between neighboring data and intensity non-uniformity artifact compensation. This chapter illustrates the use of spatial fuzzy c-means (SFCM), so as to incorporate the spatial neighboring information into the standard fuzzy c-means for pothole detection on pavement surfaces.

3.1 Fuzzy c-means clustering with spatial constraints

FCM is an unsupervised fuzzy clustering algorithm. The conventional clustering algorithms determine the "hard partition" of a given dataset based on certain criteria that evaluates the goodness of partition, so that each datum belongs to exactly one cluster of the partition. The soft clustering on the other hand finds the

"soft partition" of a given dataset. And in "soft partition", the datum can partially belong to multiple clusters. Soft clustering algorithms do generate a soft partition that also forms fuzzy partition. A type of soft clustering of special interest is one that ensures membership degree of point x_j in all clusters adding up to one (Eq. (6)), and also satisfies the constrained soft partition condition.

$$\sum_i \mu_{c_i}(x_j) = 1, \forall x_j \in X \tag{6}$$

The fuzzy c-means is a clustering method which allows one piece of data to belong to two or more clusters [28, 29]. The standard FCM algorithm considers the clustering as an optimization problem where an objective function must be minimized, and assigns pixels to each category by using fuzzy memberships. If $I = \{x_j \in R^d\}_{j=1, ..., N}$ is a $p \times N$ data matrix, where, p represents the dimension of each x_j "feature" vector, and N represents the number of feature vectors (pixel numbers in the image), then the FCM algorithm is an iterative optimization that iteratively minimizes the objective function, with respect to fuzzy membership $'U'$, and set of cluster centroids, $'V'$ as in Eq. (7).

$$J_{FCM} = \sum_{j=1}^{N} \sum_{i=1}^{c} u_{ij}^m \cdot \left\| x_j - v_i \right\|^2 \tag{7}$$

where u_{ij} represents the fuzzy membership of pixel x_j in the ith cluster and $u = (u_1, u_2, ..., u_c)$ are the set of cluster centers; $'C'$ is the number of clusters; v_i is the ith cluster center; $\|\cdot\|$ is a Euclidean distance or the norm metric, and m is a constant for fuzziness exponent. The parameter m controls the fuzziness of the resulting partition or the fuzziness of the consequential partition, and $m = 2$ is used in this study.

The cost function is minimized when pixels close to the centroid of their clusters are assigned high membership values, and low membership values are assigned to pixels with data far from the centroid. The membership function represents the probability that a pixel belongs to a specific cluster. In the FCM algorithm, the probability is dependent solely on the distance between the pixel and each individual cluster center in the feature domain. By minimizing Eq. (7) using the first derivatives with respect to u_{ij} and v_i then setting them to zero using the Lagrange method, the membership functions and cluster centers are updated by solutions of u_{ij} and the fuzzy centers v_i:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_j - v_i \right\|}{\left\| x_j - v_k \right\|} \right)^{2/(m-1)}} \tag{8}$$

and

$$v_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{N} u_{ij}^m} \tag{9}$$

Starting with an initial guess for each cluster center, the FCM converges to a solution for v_i representing the local minimum or a saddle point of the cost function. Convergence can be detected by comparing the changes in the membership

function or the cluster center at two successive iteration steps. In an image, as illustrated in [1], the neighboring pixels are normally highly correlated. This is because these neighboring pixels possess similar feature values, and the probability that they belong to the same cluster is often high. The introduction of the spatial information is an important cue in resolving the *mixel* problem within a pavement pothole *voxel*. While this spatial relationship is important in clustering, it is not utilized in a standard FCM algorithm. To overcome the effect of noise in the segmentation process, [30] proposed spatial FCM algorithm in which spatial information can be incorporated into fuzzy membership functions directly using a spatial function. The spatial information is introduced while updating the membership function u_{ij} in the repetitive FCM algorithm because the neighborhood pixels possess same properties as the center pixel. To exploit the spatial information, the spatial function is defined by h_{ij} (Eq. (10)).

$$h_{ij} = \sum_{k \in NB(x_j)} u_{ik} \tag{10}$$

where $NB(x_j)$ is a local square window centered on pixel x_j in the spatial domain, and in this illustration, a 5×5 window is used.

Like the membership function, the spatial function h_{ij} represents the probability that pixel x_j belongs to the *i*th cluster. The spatial function of a pixel for a cluster is large if the majority of its neighborhood belongs to the same clusters. The spatial function is used in updating the membership function again, and is incorporated into membership function as follows as presented in Eq. (11) [30].

$$u'_{ij} = \frac{u_{ij}^p h_{ij}^q}{\sum\limits_{k=1}^{c} u_{kj}^p h_{kj}^q} \tag{11}$$

where p and q are two parameters used to control the relative importance of both the membership and spatial functions respectively.

In a homogenous region within an image, the spatial functions will strengthen the original membership, and the clustering result remains unchanged. However, for a noisy pixel, this formula reduces the weighting of a noisy cluster by the labels of its neighboring pixels. As a result, misclassified pixels from noisy regions or spurious blobs can easily be corrected. The spatial FCM with parameter p and q is denoted $SFCM_{p,q}$. For $p = 1$ and $q = 0$, the $SFCM_{1,0}$ is identical to the conventional or standard FCM. In the $SFCM_{p,q}$, the objective function is not changed, instead the membership function is updated twice. The first update is the same as in standard FCM that calculates the membership function in the spectral domain. However in the second phase, the membership information of each pixel is mapped to the spatial domain, and the spatial function is computed from that. The spatial function is defined as the sum of the membership values in spatial domain in the entire neighborhood around the pixel under consideration. The FCM iteration proceeds with the new membership that is incorporated with the spatial function. The iteration is stopped when the maximum difference between two cluster centers at two successive iterations is less than a threshold (=0.02). After the convergence, defuzzification is applied to assign each pixel to a specific cluster for which the membership is maximal. The $SFCM_{p,q}$ works well for high as well as low density noise, and can be applied for single and multiple feature data. As compared to other methods FCM based methods, $SFCM_{p,q}$ gives superior results without any boundary leakage even at high density noise, when the q value is carefully selected [31].

3.2 Depth image data smoothing and hole-filling

To correctly analyze and potentially combine the RGB image with the depth data, the spatial alignment of the RGB and the depth camera outputs is necessary. Additionally, the raw depth data are very noisy and many pixels in the image may have no depth due to multiple reflections, transparent objects or scattering in certain nearby surfaces. As such the inaccurate and or missing depth data (holes) need to be recovered prior to data processing. The recovery is conducted through application-specific camera recalibration and or depth data filtering. In this section we deal with the depth data filtering first, and in the next subsection, the camera calibration is discussed. By enhancing the depth image using color image, the following issues are addressed: (i) due to various environmental reasons, specular reflections, or simply the device range, there are regions of missing data in the depth map; (ii) the accuracy of the pixels values in the depth image is low, and the noise level is high. This is true mostly along depth edges and object boundaries, which is exactly where such information is most valuable; (iii) despite the calibration, the depth and color images are still not aligned well enough. They are acquired by two close, but not similar, sensors and may also have differences in their internal camera properties (e.g., focal length). This misalignment leads to small projection differences, even, again, these small errors are more noticeable especially along edges, and (iv) usually the depth image has lower resolution than the color image, and therefore it should be up-sampled in a consistent manner.

Because of the limitations in the depth measuring principle and object surface properties, the depth image from Kinect inevitably contains optical noise and unmatched edges, together with holes or invalid pixels, which makes it unsuitable for direct application [32]. In order to remove noise from the depth image, the joint bilateral filter is preferred. This is because the joint bilateral filter has the advantage of preserving edges while removing noises, analyzing through every image pixel and replacing every image pixel-by-pixel with the median of the pixels in the corresponding filter region R. This process can be expressed according to Eq. (12).

$$I'(u,v) \rightarrow median\{I(u+i,v+j)|(i,j) \in R\} \tag{12}$$

where, (u,v) is the position of the image pixel and (i,j) is the neighborhood size of the image region and these are specified as a two element numeric vector of positive integers. By using median filtering, each output pixel contains the median value in the $i \times j$ neighborhood around the corresponding pixel in the input image.

In filling holes in depth images: (i) [33] used bilateral filter and median filter in the temporal domain; (ii) [34] proposed joint bilateral filter and Kalman filter for depth map smoothing, and to reduce the random fluctuations in the time domain. Jung [35] proposed a modified version of the joint trilateral filter (JTF) by using both depth and color pixels to estimate a filter kernel and by assuming the presence of no holes. Liu et al. [36] employed an energy minimization method with a regularization term to fill the depth-holes and remove the noise in depth images. The linear regression model utilized was based on both depth values and pixel colors. From the above studies, it is noted that the methods are primarily based on different types of filters to smooth noise in depth images and to fill holes by using color images to guide the process.

Introduced by [37], the bilateral filter is a robust edge-preserving filter with two filter kernels: a spatial filter kernel and a range filter kernel, which are traditionally based on a Gaussian distribution, for measuring the spatial and range distance between the center pixel and its neighbors, respectively [38].

By letting I_X be the color at pixel $\mathbf{x}$, and I_X^I be the filtered value, it is desired for I_X^I to be:

$$I_X^I = \frac{\sum_{y \in N(x)} f_S(x,y) \cdot f_R(I_x, I_y) \cdot I_y}{\sum_{y \in N(x)} f_S(x,y) \cdot f_R(I_x, I_y)} \qquad (13)$$

where $\mathbf{y}$ is a pixel in the neighborhood $N(\mathbf{x})$ of pixel $\mathbf{x}$, where $f_S(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma_S^2}\right)$ and, $f_R(I_x, I_y) = \exp\left(-\frac{\|I_x - I_y\|^2}{2\sigma_R^2}\right)$ are the spatial and range filter kernels measuring the spatial and range/color similarities. The parameter σ_S defines the size of the spatial neighborhood used to filter a pixel, and σ_R controls how much an adjacent pixel is down-weighted because of the color difference.

The limitation of the conventional bilateral filter is that it can interpret impulse noise spikes as forming an edge. A joint or cross bilateral filter [39, 40] is similar to the conventional bilateral filter except that in the case of the joint bilateral filter, the range filter kernel $f_R(\cdot)$ is computed from another image called the *guidance image*. The guide image J indicates where similar pixels are located in each neighborhood. With J as the guidance image, then the joint bilateral filtered value at pixel $\mathbf{x}$ is determined as in Eq. (14).

$$I_X^J = \frac{\sum_{y \in N(x)} f_S(x,y) f_R(J_x, J_y) I_y}{\sum_{y \in N(x)} f_S(x,y) f_R(J_x, J_y)} \qquad (14)$$

It is important to note that the joint bilateral filter ensures the texture of the filtered image I^J to follow the texture of the guidance image J. In the implementation this paper, the image intensity was normalized such that it ranges from [0, 1], and image coordinates were also normalized so that $\mathbf{x}$ and $\mathbf{y}$ also reside in [0, 1].

With this depth hole filling based on the bilateral filter, the depth value at each pixel in an image is replaced by a weighted average of depth values from nearby pixels. While the joint bilateral filter has been demonstrated to be very effective for color image upsampling, if it is directly applied to a depth image with a registered RGB color image as the guidance image, the texture of the guidance image (that is independent of the depth information) is likely to be introduced to the upsampled depth image, and the upsampling errors mainly reside in the texture transferring property of the joint bilateral filter [38]. Meanwhile, the median filtering operation minimizes the sum of the absolute error of the given data [41], and is much more robust to outliers than the bilateral filter. A possible solution to the "hole-filling" problem in depth imagery is to focus on the combination of the median operation with the bilateral filter so that the texture influence can be better suppressed while maintaining the edge-preserving property [42].

3.3 Calibration of RGB and IR Kinect cameras

Despite the fact that the Kinect, like other off-the-shelf sensors, has been calibrated during manufacturing, and the camera parameters are stored in the device's memory, this calibration information not accurate enough for reconstructing 3D information, from which a highly precise cloud of 3D points should be obtained. Furthermore, the manufacturer's calibration does not correct the depth distortion, and is thus incapable of recovering the missing depth [43]. Using a 9×8

checkerboard, with 30 mm square fields, a set of close-up RGB/IR images of the checkerboard placed in different positions and orientations (**Figure 3(a)**), can be collected and used for calibration. The Bouguet's Camera Calibration Toolbox [44] in MATLAB can be used for the identification of RGB and IR camera parameters, utilizing the two versions of Herrera's method [45]. IR camera calibration, the IR emitter should be disabled during imaging so as to achieve appropriate light conditions. The output matrices for the intrinsic, distortion and extrinsic calibration parameters are presented in **Table 2**.

3.3.1 Initialization of intrinsic and extrinsic calibration

For the color camera, the initial estimation of I_c and $T_c^{(i)}$ for all calibration images is carried out as described in Bouguet's toolbox. The intrinsic parameters for the depth camera are defined as $I_d' = \{f_d, c_d, k_d, c_0, c_1\}$, since the depth distortion terms are not considered. They are initialized using preset values, which are publicly
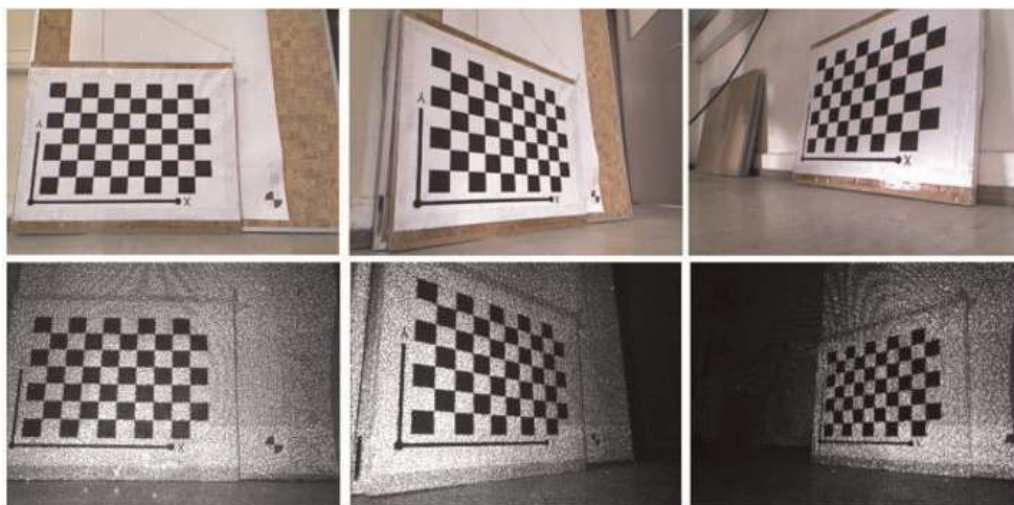
Figure 3.
Checkerboard RGB (top) images and the corresponding IR (bottom) calibration images. From the case study roads, a database of 10,540 color and depth test image frames has been acquired and being processed.

Intrinsic calibration matrix			
536.782668	0.000000	319.133028	
0.000000	536.889190	258.356500	
0.000000	0.000000	1.000000	
Distortion calibration matrix			
0.243645	−0.572745	−0.008210	0.000119
Extrinsic calibration matrix			
0.999987	−0.004894	−0.001283	110.506445
−0.004661	−0.989735	0.142836	−133.830468
−0.001969	−0.142828	−0.989746	867.124291
0.000000	0.000000	0.000000	1.000000

Table 2.
Intrinsic, distortion and extrinsic calibration matrix parameters.

available for the Kinect, online. For each input disparity map i, the plane corners are extracted, defining a polygon. For each point x_d inside the polygon, the corresponding disparity d is used for computing a depth value z_d using $z = \frac{1}{c_1 d_u + c_0}$, where $d = d_u$ since the measured disparities are used, and c_0 and c_1 are part of the depth camera's intrinsics. The correspondences (x_d, y_d, z_d) are used for computing 3D X_c points originating a 3D point cloud. To each 3D point cloud, a plane is fitted using a standard total least squares algorithm.

3.4 Pothole search engine

As a pre-processing step and prior to the segmentation and clustering of the RGB and depth data, pothole search engine (PSE) is necessary. It is then possible to extract potholes-only images for further autonomous processing. This can be accomplished by using a 2-class k-means clustering of the candidate RGB image frames, and is confirmed using ellipsoidal fitting on the classified binary image frame.

3.4.1 k-*means clustering and edge ellipse fitting for pothole search*

Since the data collected comprises of pothole and non-pothole pavement defect image frames, the first preprocessing step after the calibration is to eliminate the non-pothole images from the database. Using unsupervised classification on the acquired RGB data frames, images with potential potholes are selected based on k-means clustering [46], and adaptive median filtering. From the candidate potholes images, edge lines are estimated and the corresponding ellipse(s) are fitted using least squares optimization. This algorithm is applied in a batch processing mode, and the efficiency of the approach is then confirmed by using visual inspection and comparison.

3.4.2 Horizontal and vertical integral projection (HVIP)

Integral projection (IP) has the discriminative to accumulate and resolve the pixel histograms into pothole and non-potholes pixels, by analyzing the horizontal and vertical (HV) pixel distributions within an image, represented by horizontal and vertical projections. Given a grayscale image $I(x, y)$, the horizontal and vertical IPs are defined as follows in Eqs. (15) and (16).

$$HP\,(y) = \sum_{i \in x_y} I(i,j) \qquad (15)$$

$$VP\,(x) = \sum_{j \in y_x} I(i,y) \qquad (16)$$

where HP and VP are the horizontal and vertical IP, respectively. x_y and y_x denote the set of horizontal pixels at the vertical pixel y and the set of vertical pixels at the horizontal pixel x, respectively.

3.4.3 Database search for candidate pothole image frames using ellipse fitting and HVIP

With a visual comparison of 99% efficiency for the pothole database search, **Table 3** shows the results using the pothole search engine (PSE). The ellipse detection indicates the presence of defect or no-defect within the image, and also defines the orientation of the pothole with respect to the longitudinal profile of the road.
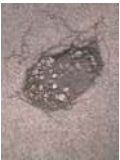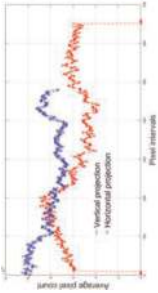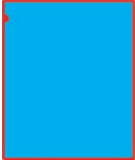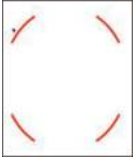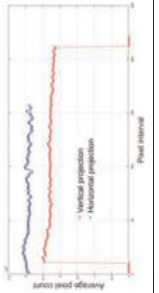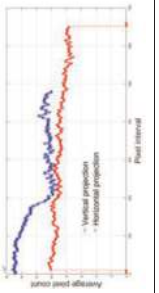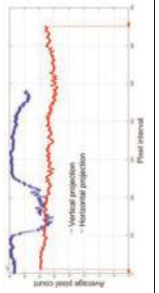
Test data	Pavement RGB image frame	2-class *k*-means classification and adaptive median-filtering	Fitted edge on candidate pothole image	Ellipse-fitting in PSE for pothole presence detection	Horizontal and vertical integral projection (HVIP) plots
#1					
#2					
#3					
#4					

Table 3.
PSE and HV-integral projection search for pothole and non-pothole frames from RGB test data.

The results of horizontal and vertical IP (HVIP) analysis for several pavement images with varied sized pixels are presented in **Table 3**. As observed from the test results, a structurally healthy pavement image with non-potholes (e.g., test image #2) is generally characterized by recognizably stable signals of both horizontal and vertical integral projections. On the other hand, the integral projections of images containing potholes (e.g., test images #1, #3 and #4), has peak(s) in either the vertical or horizontal or both IPs, depending on the strength or the severity of the pothole and lighting conditions. Where both the horizontal and vertical signals are strong, the locations of the two peaks tend to be relatively close to each other. Thus in addition to the ellipsoidal fitting, HVIP can effectively be used in the extraction of pothole and non-pothole image frames in a pothole database search engine system. In the PSE search system, data acquired under varied illumination conditions were tested, to ensure the effectiveness of the system with data of different resolutions.

4. Pothole metrology data parametrization

Figure 4 illustrates the conceptual approximation of a pothole with dimensional parameters that define the pothole metrology as: width, depth, surface area and volume. Assuming the potholes have the shape of a circular paraboloid, then in 2D they can be represented by the function $f(x,y) = x^2 + y^2$.

4.1 Pothole depth determination using depth image

The depth-image plane (**Figure 4**) is one of the noise factors, whereby the plane is not necessarily parallel to the pavement surface. The noise points, which are the non-defect points between the pavement-pothole plane and the camera, have to be filtered out for the accurate depth detection and the subsequent 2D-pothole detection from the depth image. The general principle of removing the outlier points (noise), is by determining the local minimum of each column and then subtracting from the column itself in order to extract the pothole from the rest of data [47]. The minimum of each column defines the depth below which the pothole starts on the road pavement surface, and is referred to as the depth-image plane. Using this approach, the depths d_i including the maximum depth $d_{i_{max}}$ can be quantified, and the mean depth $\overline{d_i}$ for a given pothole is also computed.
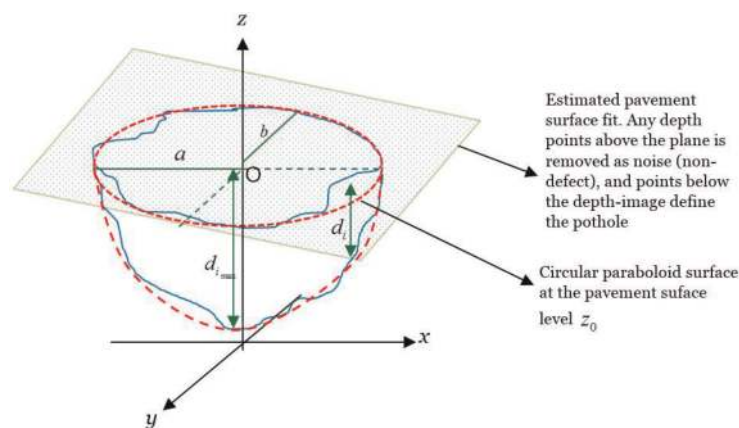
Figure 4.
Representation and approximation of pothole metrology elements: depth, width, surface area and volume.

4.2 Pothole width measurement

The width of a pothole can be defined by the semi-major a and semi-minor b axes, on the assumption that an ellipse, based on the major path elliptic regression, is used pothole shape extraction [48]. To determine the lateral width of the pothole, it can be estimated using a circular paraboloid, which is an elliptical paraboloid. And, an elliptical paraboloid is a surface with parabolic cross-sections in 2-orthogonal directions and 1-elliptical cross-section in the other orthogonal direction. Using an edge detection algorithm, the near-true shape of the pothole is first derived using the proposed SFCM, and then an elliptical fit is used to approximate the shape, from which the axes are defined for the calculation of the surface area and volume of the pothole.

4.3 Pothole surface area determination

In order to determine the surface area of the pothole, the optimally detected edge is used to fit the shape of the pothole as either elliptic paraboloid or circular paraboloid. While the former is defined by the dimensions of semi-major axis a and semi-minor axis b, the latter is defined by the estimated radius r. The surface area is then computed by using the surface integrals of either of the paraboloids [49], as respectively shown in Eqs. (17) and (18) for the elliptic and circular paraboloids.

$$A = \pi ab \tag{17}$$

$$A_r = \frac{\pi}{6}\sqrt{(1 + 4r^2)^3} \tag{18}$$

If pixels counts are used, then Eq. (19) can be implemented, [8]. Whereby in Eq. (19), l is the pixel size and I_p is the binary value of pixel at coordinate position (x,y). The area A_p is estimated on the basis of the average of a 2×2 window.

$$A_p = l^2 \cdot \sum_x \sum_y I_p(x,y) \tag{19}$$

4.4 Pothole volume estimation

According to [50], if T is a closed region bounded by a surface S, and F is a vector field defined at each point of T and on its boundary surface, then $\int\int\int_T Fdv$ is the volume integral of F through the bounded region T. As in case for the surface area of a pothole, the area is either estimated by an elliptic paraboloid or a circular paraboloid. The volume of the elliptic paraboloid V can be estimated according to Eq. (20), and the volume V_r 0f the pothole is estimated using a circular paraboloid as in Eq. (21).

$$V = \frac{4}{3}\pi abd_{\max} \tag{20}$$

$$V_r = \frac{\pi r^4}{2} \tag{21}$$

Since the depth for each pixel d_i is obtainable from the depth image, the integration of all small volumes represented by each pixel leads to the total volume of

area within the frame [51]. Therefore the estimated volume V_d in terms of the pixel depth is given by Eq. (22)

$$V_d = l_p^2 \cdot \sum_y \sum_x I_d(x,y) \cdot I_p(x,y) \tag{22}$$

where V_d is the total pothole volume, and $I_d(x,y)$ is depth of pixel p at location (x,y).

4.5 Prototype implementation strategy for pothole detection using low-cost sensor

Figure 5 illustrates the processing steps in implementing the detection, and visualization potholes and related metrological parameters from the Kinect v2.0 RGB-D, based on the experimental iMMSS data capture system. In summary the processing system should comprise of data acquisition and geometric transformation; preprocessing for noise minimization; cascaded pothole detection approach from fused RGB-D data using dual-clustering approach comprising of k-means and spatial fuzzy c-means, and a parallel processing system for pothole area and volume detection from RGB and depth imagery.
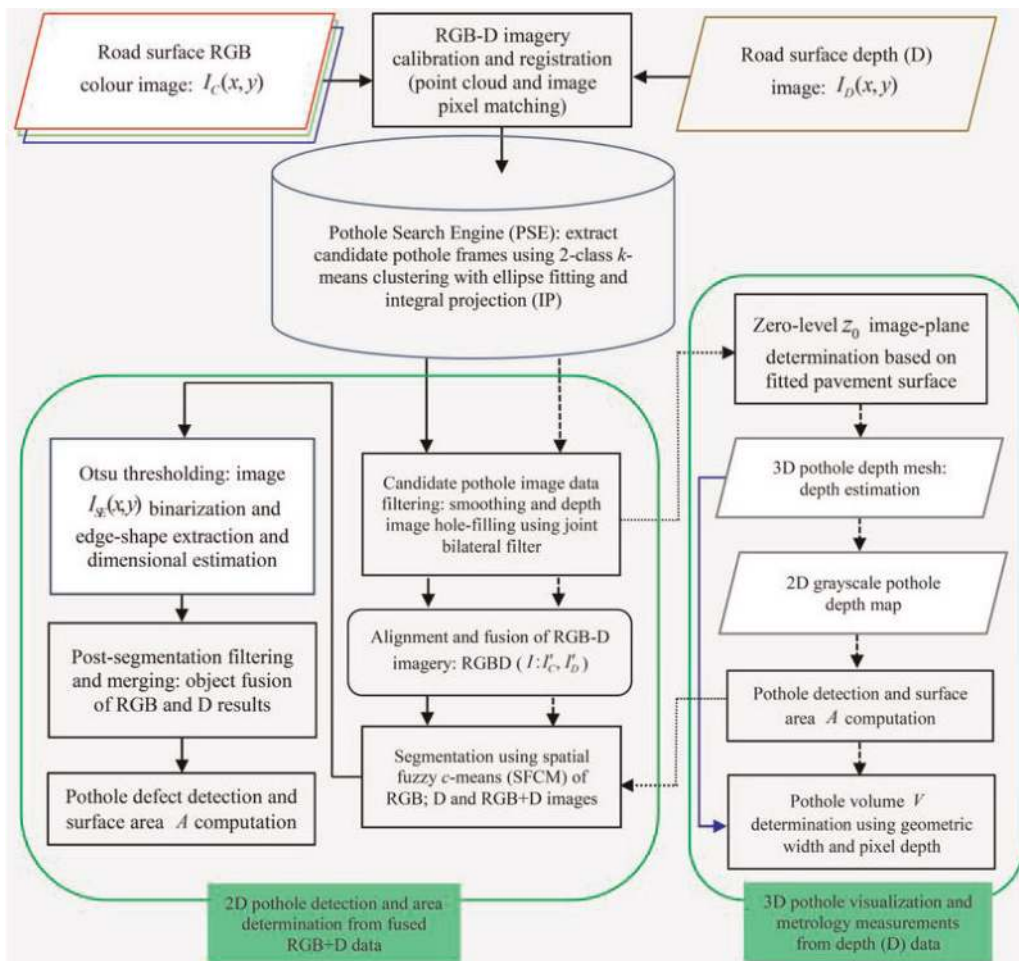
Figure 5.
Processing pipelines for pothole detection based on cascaded dual-clustering and pothole metrology quantification and visualization from multimodal iMMSS low-cost RGB-D sensor system.

5. Some experimental results and analysis

5.1 Pothole detection using SFCM segmentation

The results for the clustering of the RGB imagery using FCM and SFCM are comparatively presented. Where there is low spectral heterogeneity, the first Principal Components Transform image (PCT-band 1) is used in the FCM and SFCM clustering. The results in **Table 4** shows that the inclusion of the spatial neighborhood information using the SFCM, results in a more compact detection of the potholes, by segmenting the potholes from the non-potholes and ensuring homogeneity within the pothole itself, hence taking the spatial cues in clustering. Furthermore, the SFCM performs much better than FCM especially under different lighting conditions.
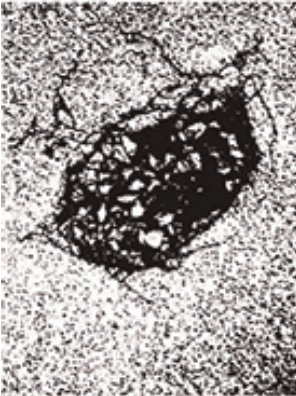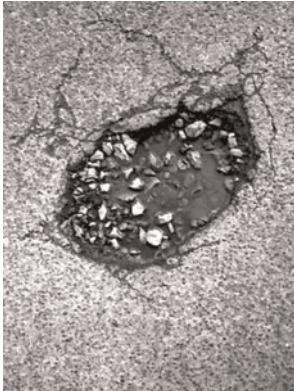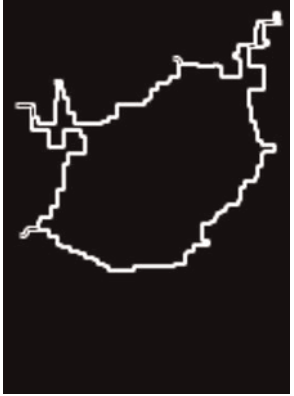
5.2 Pothole depth imagery representation
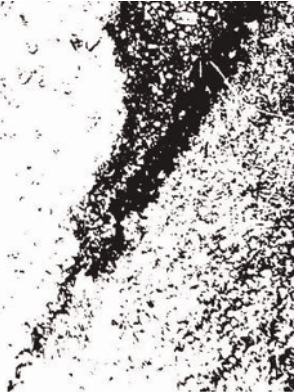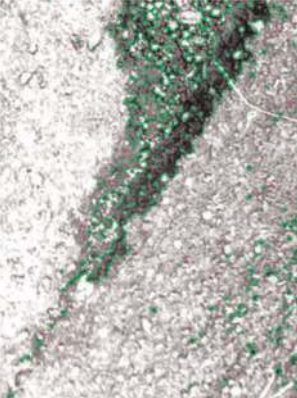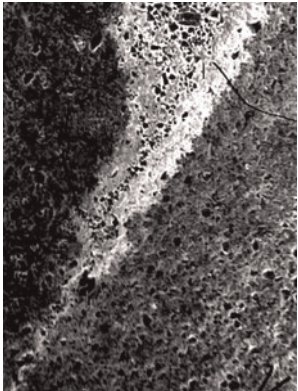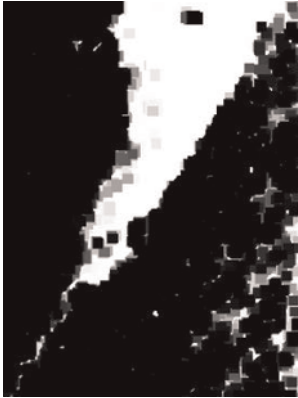
Defects on pavements are defined as surface deformations that are greater than a threshold as illustrated in **Figure 6(b)**. Since the captured depth data is corrupted with noise, the depth-image plane as illustrated in **Figure 4** (**Figures 6(b)** and **6 (c)**), is not necessarily parallel to the surface that is under inspection. This is solved by fitting a plane to the points in the depth image (**Figure 6(b)**), that are not farther than a threshold from the IR camera (**Figure 6(c)**). By using the random sample consensus (RANSAC) algorithm [52], the plane is fitted to the points, and the depth image is subtracted from the fitted plane, with the results in **Figure 6(d)**. To discriminate between the depressions (potholes) and the flat regions (non-potholes), the Otsu's thresholding algorithm is used. Sample results of the depth-image segmentation are sequentially presented in **Figure 6**.

5.3 Feature based RGB-D data fusion for enhanced pothole segmentation

In this section, an illustration on the potential of fusion of the depth and color image at the object or feature level is demonstrated. A possible two-way fusion approach comprising of either: (i) pre-pothole detection fusion involving the enhancement of the color image with the depth image, or (ii) post-pothole detection fusion of the pothole defect features as independently determined from the RGB and depth images respectively is proposed and conceptually represented in **Figure 7**. The first approach presents a joint segmentation approach, which is similar to extracting consistent layers from the image where each layer segment in terms of both color and depth. It is common for real scene object, like pavement pothole surfaces, to be characterized by different intensities and a small range of depths. The incorporation of the depth information into the segmentation process, allows for the detection of real pothole object boundaries instead of just coherent color regions, and the objective is to enhance the application relevant features in the resultant fused image product.

The potential and significance of fusion of RGB and depth imagery is illustrated in **Figures 8** and **9**, using the pothole edge identification from the RGB and depth image data. **Figure 8** shows an RGB and depth (RGB-D) single frame pavement data acquired Kinect experimental setup. The RGB is smoothened (left frame) using the median filter, while hole-filling using the joint bilateral filter is applied to the depth image (right frame). It is observed that the two images complement each other. Comparing the corrected image datasets, it is observed that the depth image clearly defines the pothole edges as compared to the fuzzy representation of the

Clustering and segmentation results for pothole detection using fuzzy *c*-means clustering

Original RGB pothole image frame and the first principal components (PC) image	FCM clustering	SFCM segments and segmentation	SFCM results with Otsu thresholding

FCM clustering

SFCM clustered segments

Detected pothole after thresholding

FCM segmentation results with Otsu thresholding

SFCM segmentation results

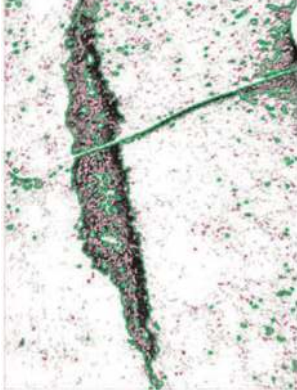Perimeter of the detected pothole

Clustering and segmentation results for pothole detection using fuzzy *c*-means clustering

Original RGB pothole image frame and the first principal components (PC) image	FCM clustering	SFCM segments and segmentation	SFCM results with Otsu thresholding

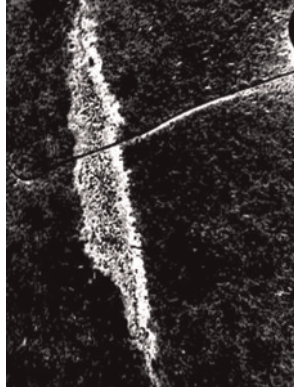SFCM results with Otsu thresholding

Detected pothole after thresholding

Perimeter of the detected pothole

SFCM segments and segmentation

SFCM clustered segments

SFCM segmentation results

FCM clustering

FCM clustering

FCM segmentation results with Otsu thresholding

Original RGB pothole image frame and the first principal components (PC) image

Original RGB pothole image frame and the first principal components (PC) image	Clustering and segmentation results for pothole detection using fuzzy *c*-means clustering		
	FCM clustering	**SFCM segments and segmentation**	**SFCM results with Otsu thresholding**

FCM clustering

SFCM clustered segments

Detected pothole after thresholding

FCM segmentation results with Otsu thresholding

SFCM segmentation results

Perimeter of the detected pothole

Table 4.
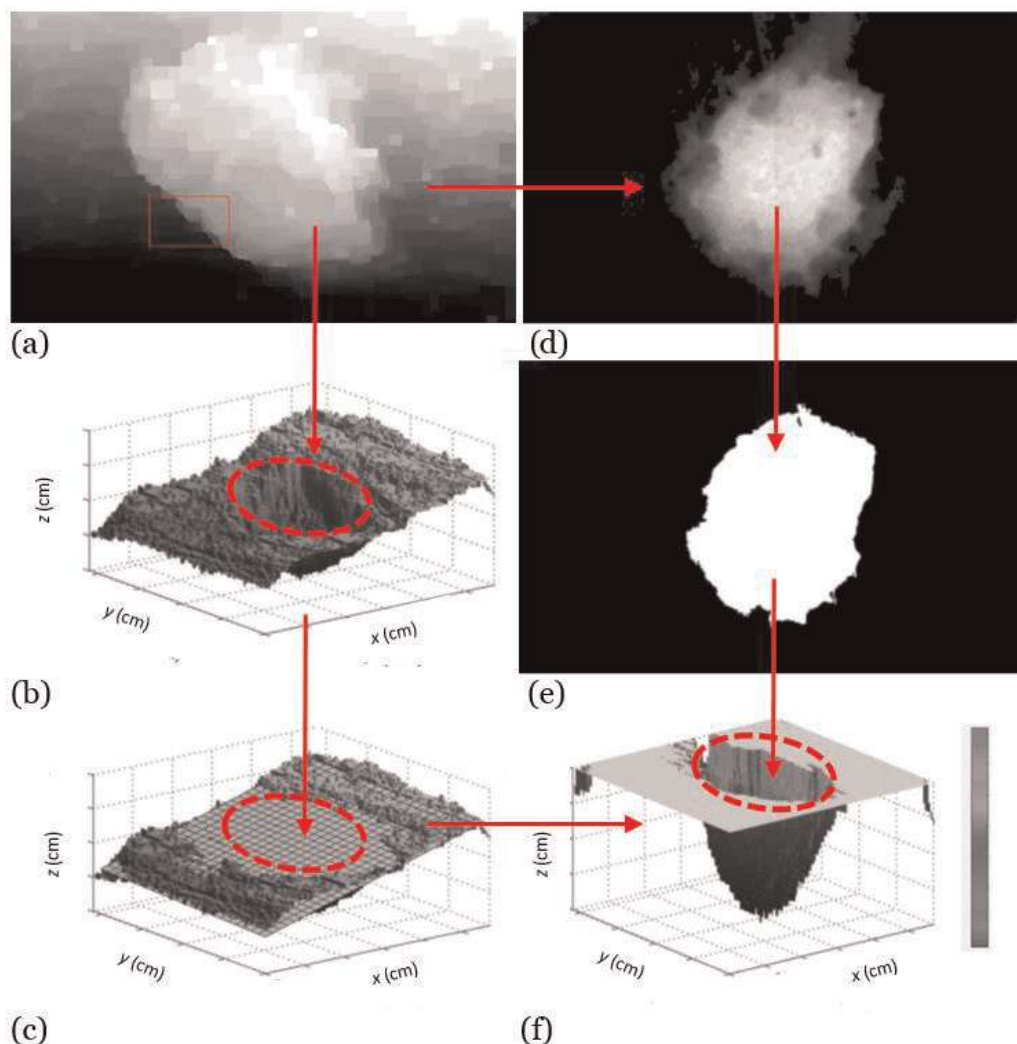*Pothole detection results using FCM and SFCM.*

Figure 6.
(a) Pothole depth image. (b) Corresponding depth data to RGB image in (a). (c) Plane fitting using RANSAC [52]. (c) Relative depth obtained from subtracting the depth values from the fitted plane. (d) Rotated gray-scale representation of the relative depth values. (e) Detected pothole defect obtained from binarizing image (d) using the Otsu's thresholding. (f) Depth map of the detected pothole with dimensions in millimeters (cm).

edges by the color image (**Figure 9**). This implies that it is possible to improve the pothole detection from RGB imagery through fusion of the RGB and depth image datasets (feature fusion) or through post-segmentation fusion (object fusion). For this chapter, only a discussion and potential illustration is presented.

5.4 Evaluation of results and quantification of pothole metrology parameters

An evaluation of the low-cost pavement pothole detection system is carried out using 55 depth image frames comprising of 35 images with potholes and 20 defect-free frames were evaluated. The results of the illustrative evaluation are presented in **Tables 5** and **6**, respectively in terms of the confusion matrix and the overall performance indices: TP, TN, FP, and FN which respectively represent the true positive, true negative, false positive and false negative. In **Table 6**, accuracy is defined as the proportion of the true classifications in the test dataset, while precision is the proportion of true positive classifications against all positive
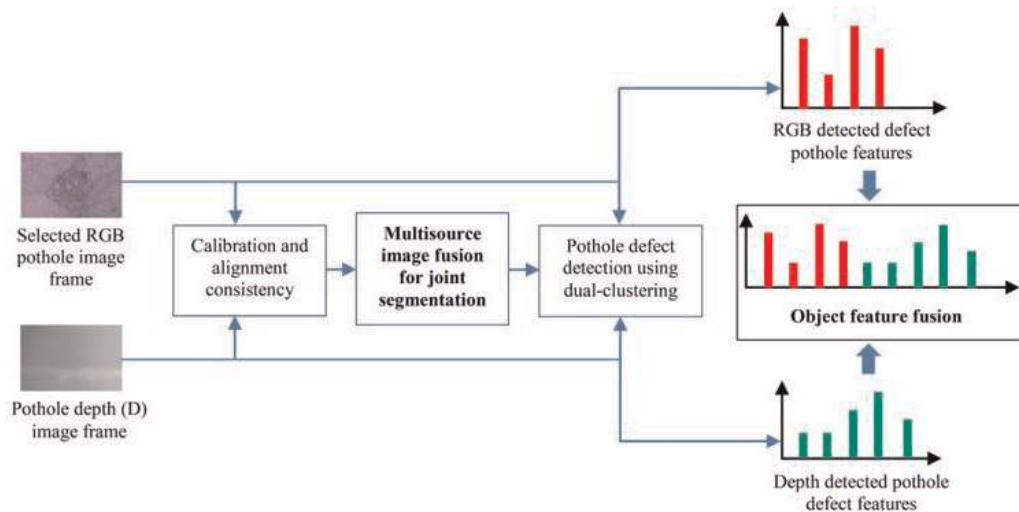
Figure 7.
Conceptual framework for the RGB-D pothole defect detection based on pre-detection image feature fusion and post-detection object fusion.
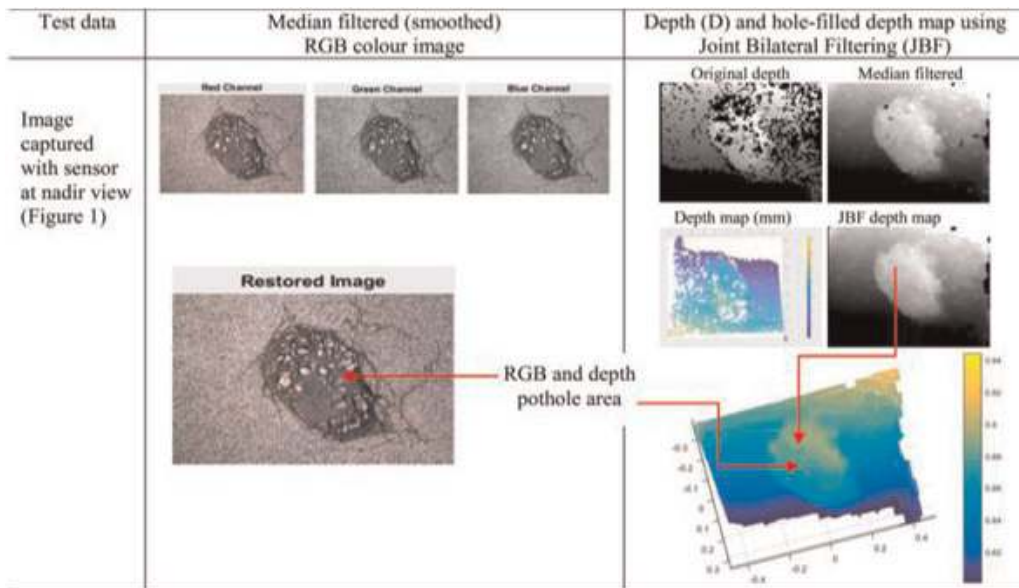

Figure 8.
Comparing RGB imagery (a) and filtered depth map for pothole and non-pothole mapping on asphalt pavement.

classifications. The overall results show that the detection rate for potholes was at 82.8% degree of accuracy.

In terms of the pothole metrology measurements, **Table 7** presents a sample summary of the results for the metrologic data quantification as characterized by: length and width, mean depth, mean surface area and volume of the potholes within image frames, and the resulting relative errors. From the results in **Table 7**, it is observed that while for some pothole defects the estimated dimensions are close to the ground-truth manual measurements, in few cases i.e., less 25% of the images, the relative error is more than 20%. This observed error magnitude in the pothole-detection system was attributed to the shape and edge complexity of the potholes, which are mathematically complex to represent and estimate appropriately and accurately as demonstrated in **Figure 6**.
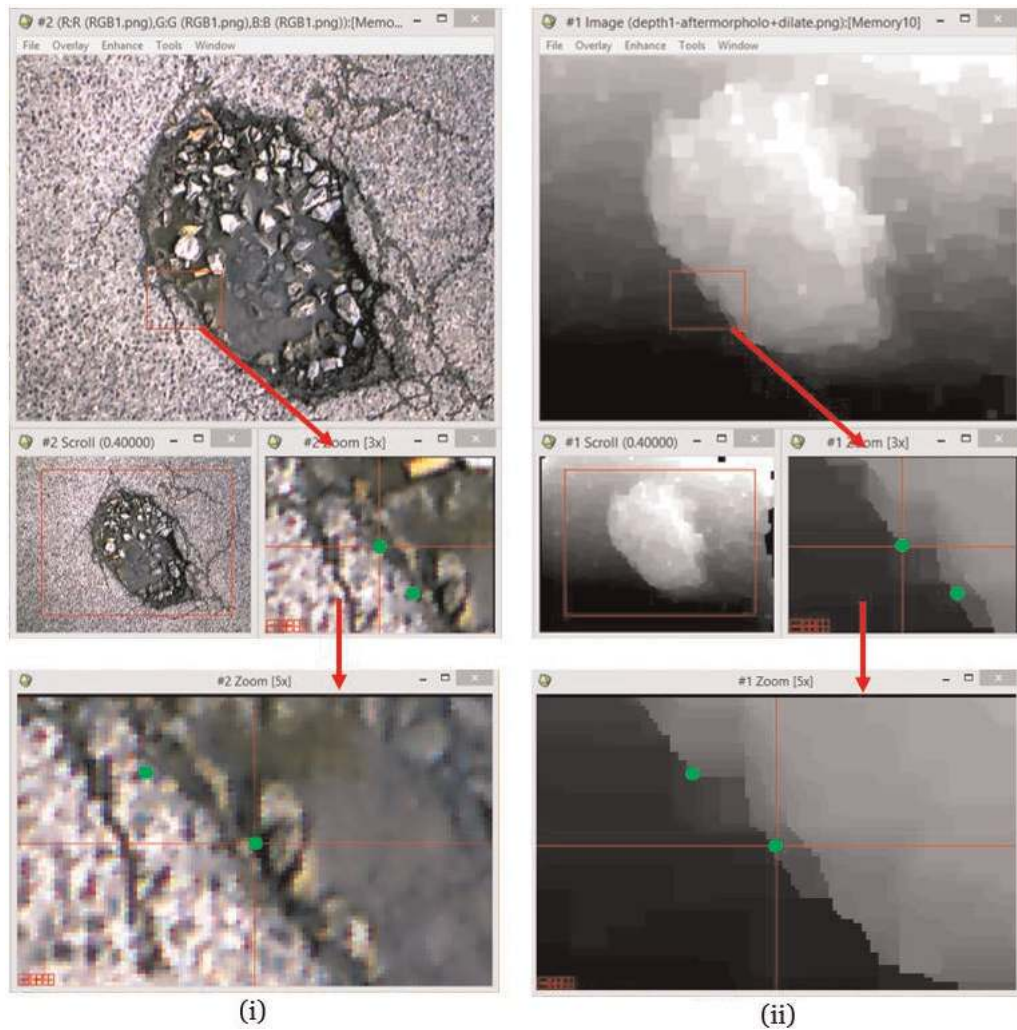
Figure 9.
Illustration of the significance of depth in pothole edge mapping in relation to pothole data fusion and improved detection. (i) RGB image. (ii) Depth map.

Prediction	Ground truth	
Classified	Defective	Defect-free
Defective (potholes)	TP = 40	FP = 5
Defect-free (non-potholes)	FN = 15	TN = 50

Table 5.
Confusion matrix of the evaluated pothole-defect detection system.

Index	Accuracy (%)	Precision (%)	Recall (%)
Value	82.8	88.8	72.7

Table 6.
Overall performance of the pothole-defect detection system.

Defect ID#	Ground-truth		Proposed method		Relative error		Proposed iMMSS method		
	Length (cm)	Width (cm)	Length (cm)	Width (cm)	Length (%)	Width (%)	Mean depth (cm)	Mean area (cm²)	Volume (cm³)
1	53.5	48.8	52.2	45.4	2.43	7.00	4.4	21.38	94.072
6	26.1	17.8	29.1	13.9	11.49	28.26	5.6	27.21	152.376
11	64.4	60.1	60.9	63.4	5.43	5.49	3.8	18.46	70.148
27	45.9	47.7	42.0	46.3	8.50	2.94	.59	28.66	169.094

Table 7.
Sample comparison of detected pothole metrological parameters with ground-truth measurements.

6. Conclusions

This chapter presents a robust approach for cost-effect detection of potholes on asphalt pavements. By first proposing a system for pavement surface mapping using Kinect v2.0 and based on the iMMSS hardware-software system, the implementation first incorporates k-means clustering and horizontal-vertical integration as data search or filtering algorithms, followed with spatial fuzzy c-means (SPCM) segmentation for pothole and non-pothole detection. The results of the processing illustrates the potential of using RGB and depth image in the detection of potholes based on low-cost consumer grade sensors, and shows the potential of fusing RGB + depth data for improved pothole detection.

From the experimental analysis, it is conclusive that using a single Kinect may not only limit the maximum traveling speed for data collection, but does not also cover the whole width of a traffic lane. This means that the field of view (FOV) can be increased by determining and using an array of Kinect sensors so that the lateral data collection extent can be increased. Further, the development of suitable depth and RGB fusion should be investigated both at object and at feature fusion levels.

In summary, it is demonstrated that low-cost and high-performance vision and depth sensors are capable of providing new possibilities for achieving autonomous inspection of pavement structures, and are suitable for overcoming the spatial and temporal limitations associated with both the manual human-based inspection and the expensive techniques. Overall, the findings of the study are significant, in terms of the new data and their processing challenges and results.

Acknowledgements

Author details

Yashon Ombado Ouma
Department of Civil Engineering, Geomatics Section, University of Botswana,
Gaborone, Botswana

*Address all correspondence to: yashon.ouma@gmail.com

IntechOpen

References

[1] Ouma YO, Hahn M. Wavelet-morphology based detection of incipient linear cracks in asphalt pavements from RGB camera imagery and classification using circular radon transform. Advanced Engineering Informatics. 2016;**30**(3):481-499

[2] Yan WY, Yuan X-X. A low-cost video-based pavement distress screening system for low-volume roads. Journal of Intelligent Transportation Systems: Technology, Planning, and Operations. 2018;**22**(5):376–389

[3] Huang J, Liu W, Sun X. A pavement crack detection method combining 2D with 3D information based on Dempster-Shafer theory. Computer-Aided Civil and Infrastructure Engineering. 2014;**29**(4):299-313

[4] Schnebele E, Tanyu BF, Cervone G, Waters N. Review of remote sensing methodologies for pavement management and assessment. European Transportation Research Review. 2015;**7**(2):1-19

[5] Zakeri H, Nejad FM, Fahimifar A, Torshizi AD, Zarandi MHF. A multi-stage expert system for classification of pavement cracking. In: IFSA World Congress and NAFIPS annual meeting (IFSA/NAFIPS), 2013 Joint. 2013

[6] Adu-Gyamfi Y, Okine NA, Garateguy G, Carrillo R, Arce GR. Multiresolution information mining for pavement crack image analysis. Journal of Computing in Civil Engineering. 2011;**26**(6):741-749

[7] Jahanshahi MR, Kelly JS, Masri SF, Sukhatme GS. A survey and evaluation of promising approaches for automatic image-based defect detection of bridge structures. Structure and Infrastructure Engineering. 2009;**5**(6):455-486

[8] Ouma YO, Hahn M. Pothole detection on asphalt pavements from 2D-colour pothole images using fuzzy *c*-means clustering and morphological reconstruction. Automation in Construction. 2017;**83**:196-211

[9] Zakeri H, Nejad FM, Fahimifar A. Image based techniques for crack detection, classification and quantification in asphalt pavement: A review. Archives of Computational Methods in Engineering. 2017;**24**(4):935-977

[10] Zhou J, Huang PS, Chiang F-P. Wavelet-based pavement distress detection and evaluation. Optical Engineering. 2006;**45**(2):027007

[11] Cord A, Chambon S. Automatic road defect detection by textural pattern recognition based on AdaBoost. Computer-Aided Civil and Infrastructure Engineering. 2012;**27**(4):244-259

[12] Werro P, Robinson I, Benbow E, Wright A. SCANNER accredited surveys on local roads in England – accreditation, QA and audit testing – annual report 2009–10. Wokingham, Berkshire: Transportation Research Laboratory; 2010

[13] Siddiqui AA. A new inspection method based on RGB-D profiling [MSc thesis]. Blacksburg, Virginia: Virginia Polytechnic Institute and State University; 2015

[14] Pöhlmann STL, Harkness EF, Taylor CJ, Astley SM. Evaluation of Kinect 3D sensor for healthcare imaging. Journal of Medical and Biological Engineering. 2016;**36**(6):857-870

[15] Pagliari D, Pinto L. Calibration of Kinect for Xbox one and comparison between the two generations of Microsoft sensors. Sensors. 2015;**15**:27569-27589

[16] Sell J, O'Connor P. The Xbox one system on a chip and kinect sensor. IEEE Micro. 2014;**2**:44-53

[17] Kolb A, Barth E, Koch R, Larsen R. Time-of-flight sensors in computer graphics. Computer Graphics Forum. 2010;**29**(1):141–159

[18] Mutto CD, Zanuttigh P, Cortelazzo GM. Time-of-flight Cameras and Microsoft Kinect. Springer Briefs in Electrical and Computer Engineering. New York: Springer-Verlag; 2012. Ch. 2. p. 21

[19] Shotton J, Sharp T, Kipman A, Fitzgibbon A, Finocchio M, Blake A, et al. Real-time human pose recognition in parts from single depth images. Communications of the ACM. 2013; **56**(1):116-124

[20] Khoshelham K, Elberink O. Accuracy and resolution of kinect depth data for indoor mapping applications. Sensors: Journal on the Science and Technology of Sensors and Biosensors. 2012;**12**(2):1437-1454

[21] Murthy SBS, Varaprasad G. Detection of potholes in autonomous vehicle. IET Intelligent Transport Systems. 2014;**8**(6):543-549

[22] Koch C, Jog G, Brilakis I. Automated pothole distress assessment using asphalt pavement video data. Journal of Computingin Civil Engineering. 2013; **27**(4):370-378

[23] Lokeshwor H, Das LK, Sud SK. Method for automated assessment of potholes, cracks and patches from road surface video clips. Procedia—Social and Behavioral Sciences. 2013;**104**: 312-321

[24] Parker R. Algorithms for Image Processing and Computer Vision. 2nd ed. Vol. 2011. New York, NY: John Wiley & Sons, Inc; 2011

[25] Wang P, Hu Y, Dai Y, Tian M. Asphalt pavement pothole detection and segmentation based on wavelet energy field. Mathematical Problems in Engineering. 2017;**2017**; Article ID 1604130, 13 pages. DOI: 10.1155/2017/1604130

[26] Zhang J, Hu J. Image segmentation based on 2D Otsu method with histogram analysis. Proceedings of Computer Science and Software Engineering, CSSE 2008. 2008;**6**: 105-108

[27] Xu D, Zhao P, Gui W, Yang C, Xie Y. Research on spectral clustering algorithms based on building different affinity matrix. In: 25th Chinese Control and Decision Conference (CCDC); 25–27 May. Vol. 2013. 2013. pp. 3160-3165

[28] Dunn JC. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. Journal of Cybernetics. 1973; **3**(3):32-57

[29] Bezdek JC. Pattern Recognition with Fuzzy Objective Function Algorithms. MA, USA: Kluwer Academic Publishers Norwell; 1981

[30] Chuang K-S, Tzeng H-L, Chen S, Wu J, Chen T-J. Fuzzy C-means clustering with spatial information for image segmentation. Journal of Computerized Medical Imaging and Graphics. 2006;**30**:9-15

[31] Choudhry MS, Kapoor R. Performance analysis of fuzzy C-means clustering methods for MRI image segmentation. Procedia Computer Science. 2016;**89**:749-758

[32] Chen L, Lin H, Li S. Depth image enhancement for Kinect using region growing and bilateral filter. In: IEEE 21st International Conference on Pattern Recognition (ICPR). Vol. 2012. 2012. pp. 3070-3073

[33] Matyunin S, Vatolin D, Berdnikov Y, Smirnov M. Temporal filtering for depth maps generated by kinect depth camera. In: 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video. 2011. pp. 1-4

[34] Camplani M, Salgado L. Efficient Spatio-temporal hole filling strategy for kinect depth maps. Proceedings of SPIE. 2012;**82900E**:2012

[35] Jung SW. Enhancement of image and depth map using adaptive joint trilateral filter. IEEE Transactions on Circuits and Systems for Video Technology. 2013;**23**(2):258-269

[36] Liu S, Wang Y, Wang J, Wang H, Zhang J, Pan C. Kinect depth restoration via energy minimization with TV21regularization. In: Proc. IEEE International Conference on Image Processing. 2013. pp. 724-724

[37] Tomasi C, Manduchi R. Bilateral filtering for gray and color images. In: Proceedings of the 6th IEEE International Conference in Computer Vision. 1998. pp. 839-846

[38] Yang Q, Ahuja N, Yang R, Tan K-H, Davis J, Culbertson B, et al. Fusion of median and bilateral filtering for range image upsampling. IEEE Transactions on Image Processing. 2013;**22**(12):4841-4852

[39] Petschnigg G, Agrawala M, Hoppe H, Szeliski R, Cohen M, Toyama K. Digital photography with flash and no-flash image pairs. ACM Transactions on Graphics. 2004;**23**(3): 664-672

[40] Eisemann E, Durand F. Flash photography enhancement via intrinsic relighting. ACM Transactions on Graphics. 2004;**23**(3):673-678

[41] Huber P. Robust Statistics. New York, NY, USA: Wiley; 1981

[42] Khoshelham K. Automated localization of a laser scanner in indoor environments using planar objects. In: Proceedings of International Conference on Indoor Positioning and Indoor Navigation (IPIN); 15–17 September 2010; Zürich, Switzerland. 2010

[43] Su PC, Shen J, Xu W, Cheung SC, Luo Y. A fast and robust extrinsic calibration for RGB-D camera networks. Sensors. 2018;**18**:235

[44] Bouguet J-Y. Camera Calibration Toolbox for Matlab. 2015. Available from: http://www.vision.caltech.edu/ -bouguetj/calib_doc

[45] Herrera D, Kannala CJ, Heikkilä J. Joint depth and color camera calibration with distortion correction. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012;**34**(10):2012

[46] Quintanilla-Dominguez J, Ojeda-Magaña B, Cortina-Januchs MG, Ruelas R, Vega-Corona A, Andina D. Image segmentation by fuzzy and possibilistic clustering algorithms for the identification of microcalcifications. Sharif University of Technology Scientia Iranica. 2011;**18**:580-589

[47] Moazzam I, Kamal K, Mathavan S, Usman S, Rahman M. Metrology and visualization of potholes using the Microsoft Kinect sensor. In: Proceedings of the 16th International IEEE Annual Conference on Intelligent Transportation Systems. Vol. 2013. 2013. pp. 1284-1291

[48] Koch C, Brilakis I. Pothole detection in asphalt pavement images. Advanced Engineering Informatics. 2011;**25**(3): 507-515

[49] Goldstein LJ, Lay DC, Schneider DI, Asmar NH. Calculus and its Applications. London: Pearson Education International; 2007

[50] Lyons L. Mathematics for Science Students. UK: Cambridge University Press; 2000

[51] Jahanshahi MR, Jazizadeh F, Masri SF, Becerik-Gerber B. An unsupervised approach for autonomous pavement defect detection and quantification using an inexpensive depth sensor. Journal of Computing in Civil Engineering. 2013;**27**(6):743-754

[52] Qian X, Ye C. NCC-RANSAC: A fast plane extraction method for Noisy range data. IEEE Transactions on Cybernetics. 2014;**44**(12):2771-2783