Kent Academic Repository

Full text document (pdf)

Citation for published version

Dib, J. and Sirlantzis, K. and Howells, G. (2020) A Review on Negative Road Anomaly Detection Methods. IEEE Access . ISSN 2169-3536.

DOI

https://doi.org/10.1109/ACCESS.2020.2982220

Link to record in KAR

https://kar.kent.ac.uk/80607/

Document Version

Publisher pdf

Copyright & reuse
Content in the Kent Academic Repository is made available for research purposes. Unless otherwise stated all content is protected by copyright and in the absence of an open licence (eg Creative Commons), permissions for further reuse of content should be sought from the publisher, author or other copyright holder.

Versions of research
The version in the Kent Academic Repository may differ from the final published version.
Users are advised to check **http://kar.kent.ac.uk** for the status of the paper. **Users should always cite the published version of record.**

Enquiries
For any further enquiries regarding the licence status of this document, please contact:
researchsupport@kent.ac.uk

If you believe this document infringes copyright then please contact the KAR admin team with the take-down information provided at **http://kar.kent.ac.uk/contact.html**

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 24, 2019, accepted February 18, 2020, date of publication March 20, 2020, date of current version April 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2982220

A Review on Negative Road Anomaly Detection Methods

JIHAD DIB, KONSTANTINOS SIRLANTZIS, AND GARETH HOWELLS
School of Engineering and Digital Arts, Jennison Building, University of Kent, Canterbury CT2 7NT, U.K.

Corresponding author: Jihad Dib (jd669@kent.ac.uk)

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC), U.K., under Grant NPIF2018-2112938 (https://gtr.ukri.org/projects?ref=studentship-2112938).

ABSTRACT The main limitation to obstacle avoidance nowadays has been negative road anomalies which is the term we used to refer to potholes and cracks due to their negative drop from the surface of the road. This has for long been a limitation because of the fact that they exist in different, random and stochastic shapes. Today's technology lacks the presence of sensors capable of detecting negative road anomalies efficiently as the latter surpasses the sensor's limitations rendering the sensing technique inaccurate. A significant amount of research has been focused on the detection of negative road anomalies due to the fact that this topic is becoming a hot research topic. In this paper, the existing techniques will be reviewed. Their limitations will be highlighted and they will be assessed via certain performance indicators and via some chosen criteria which will be introduced.

INDEX TERMS Convolutional neural networks, computer vision, crack detection, deep learning, image processing, image classification, image texture analysis, machine learning algorithm, multi-layer neural networks, negative road anomalies detection, pothole detection, real-time.

I. INTRODUCTION

Negative road anomalies, the term we have used to refer to cracks and potholes due to their nature and their negative drop from the surface of the road, have long been an issue in obstacle avoidance due to both their nature and the lack of efficient sensors and algorithms which could detect them. This has been considered as one of the most challenging tasks as potholes could exist in different forms and shapes and in different scenarios each requiring a special set-up (a pothole in an up-ramp, a pothole in a down-ramp etc.). Many sensing techniques return false negatives when they relate to negative road anomalies due to the fact that negative road anomalies have variable shapes, depths, forms, and locations.

This stochastic irregularity in the nature of these anomalies presents a significant challenge to its detection for many reasons, such as the fact that these anomalies are available in public places throughout the year posing a challenge to many detectors which are limited by certain factors such as the environment surrounding the anomaly (light intensity, fog, rain etc.). In addition to the previous, these anomalies exist in random locations with different patterns and shapes which is a significant limitation to many detection techniques, not to forget the difference in depth and the fact that these anomalies could be filled with water posing an additional challenge to almost every detection technique available nowadays.

Detection of negative road anomalies is very important in order to facilitate road maintenance, provide a better experience in automatic driving, reduce the risk of accidents (collisions, falls etc.) for the disabled wheelchair users etc. It contributes immensely in widening the spectrum of automation of vehicles' navigation and in decreasing the different risks resulting from neglecting the presence of negative road anomalies such as the effect of the vibrations resulting by driving through negative anomalies which could pose some risks to the driver/user's health, the damage which could be done to the tires of the vehicle.

Research has attempted to tackle this problem testing different techniques and developing different methodologies varying between manual, semi-automated and full-automated techniques with the help of different technologies starting at bare eye test up to deep learning artificial intelligence techniques which gave birth to various algorithms and concept solutions which could be adopted in order to properly detect negative road anomalies. Vision systems which rely on image processing techniques were widely used as they

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

provided more accurate results as shown in [61] while others have experimented with different types if sensors and imaging tools. This has certainly contributed extensively in the development of a solution to the problem, but false negatives still profusely exist, limiting the usage of these techniques as a false-negative could jeopardize the whole process where the negative surface detection is needed.

The aim of this review paper is to collate the existing technologies with a view to their relative strengths and limitations in order to improve the detection of negative surfaces. Techniques will be reviewed. Their limitations will be identified and they will be assessed based on their weaknesses and with the help of some performance indicators and assessment criteria which will be introduced.

II. EXISTING NEGATIVE ROAD ANOMALIES DETECTION TECHNIQUES

Currently available technologies will be grouped into two main categories, Deep Learning techniques which rely on deep learning and neural networks in order to ensure the detection of the negative road anomalies and Non-Deep Learning Techniques which do not use deep learning order to tackle the same problem. Focus will be on the Deep Learning techniques which is becoming a hot topic in today's research studies due to the benefits made available with the help of deep learning techniques.

A. REVIEW STRATEGY AND PERFORMANCE MEASURES

For each method reviewed, a brief explanation of the technology will be provided along with an assessment of the strength and weaknesses of the technology itself according to specific criteria which should be considered when designing an autonomous vehicles navigation system so that the algorithm to be designed would benefit from the experience earned by assessing the available techniques and equipment and to provide more accurate reliable results.

Performance measures will be extracted from the papers where mentioned. These measures are Accuracy, Precision, Recall and F1-Score:

1) Accuracy: Ratio of the correctly predicted observations to the total observations:
    - Accuracy $= \frac{TP+TN}{TP+FP+FN+TN}$
2) Precision: Ratio of the observations correctly predicted divided by the total positive observations predicted:
    - Precision $= \frac{TP}{TP+FP}$
3) Recall: Or sensitivity, Ratio of the positive observations correctly predicted divided by the total observations:
    - Recall $= \frac{TP}{TP+FN}$
4) F1-Score: Average of the Precision and Recall:
    - F1-Score $= \frac{2\times(Recall \times Precision)}{(Recall+Precision)}$

The criteria to be considered in addition to the performance measures are efficiency, real-time functionality, computing power needed, amount of power needed, size of the system, mass-production eligibility.

B. EXISTING TECHNIQUES

1) DEEP LEARNING TECHNIQUES

Deep learning, according to F. Chollet is subsidiary field of machine learning artificial intelligence which relies on the succession of different "layers" which describe the representation of the model being detected. These layers gain more meaningfulness as we move along their succession. This representation via layers is achieved by using models named "neural networks" derived from the brain's neurons (neurobiology). In deep learning, these neural networks are formed of a stack of different layers.

A deep learning network is a "multistage information-distillation operation" which consists of a succession of different filtration layers purifying the information fed as input in order to achieve the desired functionality/result. This technique provides an accurate method to "learn" the information represented in the form of data. [1]

Due to the hype in Artificial Intelligence research and the increase in success rates and performance made available by this technology, along with the increased processing speed, the ease-of-access and availability of data, and the promising results and contribution to the research society many papers have discussed the detection of negative road anomalies (mainly potholes) using machine learning techniques:

a: VISIBLE LIGHT RGB CAMERA AS INPUT

1- Pereira et al.: Pereira *et al.* proposed a deep learning algorithm which relies on convolutional neural networks in order to provide a "low-cost" solution to the problem of pothole detection.

A neural network has been used in their proposed method consisting of 4 pairs of convolutional and pooling layers followed by a fully connected layer which relies on a Rectified Linear Unit (ReLU) as an activation function.

ReLU has been known to be a piecewise linear function of the form $f(x) = \max(x, 0) f(x) = \max(x, 0)$ which means that it ensures the retention of non-negative activation values only via the annulment of the "negative part" of the activation. The sigmoid function $S(x) = \frac{1}{1+e^{-x}} = \frac{1}{e^x+1} S(x) = \frac{1}{1+e^{-x}} = \frac{1}{e^x+1}$ has also been used in order to connect the fully connected layer to the output layer where S represents the neuron's output and x represents its input. Different kinds of filters have been used as "hyperparameters" in order to produce the final output. Their network has been trained using the following parameters:

- Number of epochs: 200
- Number of images in the training set: 13,244
- Number of images in the validation set: 3,250
- Batch size: 16
- Optimizer: Adam Optimizer (cost-function reduction method)
- Learning Rate: 0.0001
- $\beta1 = 0.9$
- $\beta2 = 0.999$
- epsilon $= 1e^{-8} 1e^{-8}$

- Cost-function: $C = -yi \log ai - (-1 - yi) \log(1 - ai)$ $C = -yi \log ai - (-1 - yi) \log(1 - ai)$
- Over-fitting avoidance: 20% dropout (random dropping of neurons when training)

A good result has been achieved via their method:

TABLE 1. Pereira *et al.* experiment result.

Accuracy	99.8%
Precision	100%
Recall	99.6%
F1-Score	99.6%

A good experiment result has been achieved by the Pereira *et al.* method, however, this experiment has a number of limitations:

1- No real-time detection capabilities have been provided by the method as the method can only be used on images and not videos nor real-time streams.
2- The dataset the authors formed has not been shared and the figures represented do not provide enough information to what kind of images have been used to test and validate the results obtained.
3- This technique has not been tested in low-light and extra-bright image scenarios which could significantly affect the test results.
4- In real-time scenarios, pothole detection via machine vision using deep learning only is not feasible due to the fact that many factors could affect the imaging tool, hence putting the service user at risk due to the fact that only one system will be relied on with no failsafe.

In summary, the assessment as per our criteria has been as follows:

1- Efficiency: This system is, according to the results provided, efficient in terms of performance.
2- Real-time functionality: This system cannot be used in real-time operations due to the fact that it can only be used on images.
3- Computing-power needed: This criterion can be managed as the system can be mounted on an Intel Neural Compute Stick or any other Deep Learning portable small processor. However, if the image size is to be made larger, this might cause an issue in terms of computational-power needed.
4- Amount of Power needed: This depends on the processor used and the equipment used.
5- Size of the system: As a deep learning system, this can be manipulated as per the available equipment.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case.

2- Anand et. al. Method: Anand *et. al.* have designed a method which uses deep learning method which ensures the detection of cracks and potholes via the use of a convolutional neural network and the study of the texture and spatial information as a features which will be learned.

The system's first step consisted of the use of SegNet network which has been described in [2]–[4]. SegNet has been used as a segmentation method in order to isolate the road-part of the image. This step has been followed by Canny edge detection used as a second mask. An iteration of edges dilation follows the previous step in order to connect the edges. After the combination of the masks, the region considered as candidate has been resized to 64 × 64 patches. The unwanted edges such as tree-branch and leaves shadows, vehicles and shiny light are abnormal results which have been considered as false candidates.

The authors have chosen SqueezeNet described in reference [5] rather than AlexNet described in [6] due to the face that SqueezeNet is a modified version of AlexNet with 30 times more speed and 1.4 times less size. The authors have removed the convolutional layer of SqueezeNet and incorporated their own encoding layer which adds the learning through dictionary along with the residual encoding. It uses weights to assign each descriptors to its specific codewords (K). This layer is intended to act like a pooling layer for the network.

The method uses supervised learning in order to train the network as it learns from the labeled set of instructions. The last step is a fully connected layer which classifies the data and a Softmax layer as a last layer because the classes are mutually exclusive. For the loss-function, the authors have used binary cross entropy.

Their network has been trained using the following parameters:

- Number of epochs: 20
- Batch size: 64
- Optimizer: Adam Optimizer (cost-function reduction method)
- Learning Rate: 0.00001
- Codewords K = 32
- Momentum: 0.9
- Training data: The authors have used 2 different sets in order to test their system:
    - GAPs dataset: Image Size 1920 × 1080 pixels
        - Number of images in the training set: 1,418
        - Number of images in the testing and validation set: 551
    - Zhang dataset: Image Size 3264 × 2448 pixels [7]
        - Number of images in the training set: 1.3 million
        - Number of images in the testing and validation set: 0.7 million

Their method has achieved a good result:

TABLE 2. Using GAPs dataset.

Accuracy	99.893%
F1-Score	72.14%

TABLE 3. Using ICIP (Zhang) dataset.

Accuracy	92.37%
F1-Score	93.01%

Anand *et al.* method has achieved a good experiment result, however, a number of limitations have been identified for this experiment:

1- This technique has not been tested in low-light and extra-bright image scenarios which could significantly affect the test results.
2- The environment at which the testing occurred has not been shared (time of the day, weather, location)
3- This system relies on the texture, so anything with a texture similar to the one of a pothole will be detected as a pothole
4- This system will fail when detecting potholes filled with water if the water was not clear.
5- In real-time scenarios, pothole detection via machine vision using deep learning only isn't feasible due to the fact that many factors could affect the imaging tool, hence putting the service user at risk due to the fact that we will be relying on one system only with no failsafe.

In summary, the assessment as per our criteria is as follows:

1- Efficiency: This system is partly efficient as its accuracy varies between 92.37% and 99.99%
2- Real-time functionality: This system can be used in real-time operations as it can be used on videos and images.
3- Computing-power needed: This criterion can be managed as the system can be mounted on an Intel Neural Compute Stick or any other Deep Learning portable small processor.
4- Amount of Power needed: This depends on the processor used and the equipment used.
5- Size of the system: As a deep learning system, this can be manipulated as per the available equipment.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case.

3- Gopalakrishnan et. al Method: Gopalakrishnan *et. al* have developed a method for pavement crack detection via the use of transfer-learning applied to pre-trained deep learning models. They have used the Federal Highway Administration and LTPP in the US and Canada's database in order to extract images of pavements and have prepared a total of 1056 images split into:

- Training: 760 images
- Validation: 84 images
- Testing: 212 images

This has been applied to the Keras implementation of the VGG-16 network [8] (a Deep Convolutional Neural Network consisting of 16 layers trained with ImageNet [9]) developed by the University of Oxford's Visual Geometry Group.

Their technique has started with the preprocessing of the digital images of pavements obtained in raw format and eliminating the edges in order to reduce the size of the image. Then, after labeling them, the dataset has been applied as training input to the VGG-16 network with more than one classifier in order to compare the results. The classifiers that have been used were Single Nearest Neighbour [10], Random Forest [11], Extremely Randomized Trees [12], Support Vector Machines [13], Logistic Regression [14].

In order to train their network, the authors have used the following parameters:

- For the Single NN:
  - Image Size: 224 × 224 pixels
  - Number of Neurons in Hidden Layer: 256
  - Dropout value: 0.5
  - Hidden Layer Activation: ReLU
  - Output Layer Activation: softmax
  - Image Batch Size: 32
  - Number of Epochs: 50
- All the other classifiers: standard 'scikit-learn' machine library in Python [15] which can be obtained from https://scikit-learn.org/stable/ (Accessed 30 May 2019) was used with its standard parameters.

The highest performance according to the authors has been achieved with the Single NN classifier:

TABLE 4. Gopalakrishnan *et. al* method using the single NN classifier.

Accuracy	90.0%
F1-Score	90.0%
Precision	90.0%
Recall	90.0%

The other classifiers have achieved less rates. Their results have detailed in the paper.

Gopalakrishnan *et al.* method did achieve a good experiment result, however, this experiment has a number of limitations:

1- The system was not tested in low-light/high-light conditions and with water-filled/reflective cracks.
2- The system was not tested in real-life, the results shared were based on the samples taken from the dataset only.
3- In real-time scenarios, crack detection via machine vision using deep learning only isn't feasible due to the fact that many factors could affect the imaging tool, hence putting the service user at risk due to the fact that we will be relying on one system only with no failsafe.

In summary, the assessment as per our criteria is as follows:

1- Efficiency: This system is partly efficient as its accuracy is, according to the authors, 90.0%
2- Real-time functionality: This system can be used in real-time operations as it can be used on videos and images.
3- Computing-power needed: This criterion can be managed as the system can be mounted on an Intel Neural

Compute Stick or any other Deep Learning portable small processor.

4- Amount of Power needed: This depends on the processor used and the equipment used.

5- Size of the system: As a deep learning system, this can be manipulated as per the available equipment.

6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case.

4- Suone et. al. Method: Suone *et. al.* designed a system that relies on a YoLo (You Only Look Once) version 2 as a deep learning convolutional neural network in order to detect and identify potholes.

The authors have used two different YoLo architectures, the Darknet YoLo v2 architecture [59] and their own proposed architecture which attempts to reduce the cost of computation and the size of the model of the network. The authors method requires 18 million parameters in oppose to the Darknet architecture which requires 48 million. In order to detect more than one object, the authors have used the Anchor Box Model in order to predict (5 + numberOfClass) × numberOfAnchorboxes with each box designed to detect objects with different sizes and aspect ratios. The authors also used the k-means clustering technique on their training dataset in order to obtain five unique anchor boxes with different width and height (detailed in their paper). The newly generated anchor boxes are oriented more towards the dataset they are using. The authors have also made some modifications to the existing YoLo architecture in order to generate their own architecture. This has been achieved via the following:

1- Removal of the 23rd, 24th and 29th layer saving around 30 million parameter calculations

2- Introduction of a filter of size 2048 on the 23rd layer. This has been achieved by modifying the existing architecture's 26th convolutional layer to 256 filters (used to be 64 filters)

3- Reorganizing the 24th layer's depth to 13 x 13 x 1024 in order to reorganize the 25th layer's depth to 1024

4- Routing layer 26 and 25 with the 22nd convolutional layer

5- Modifying the existing model's anchor boxes' width and height in order to create the new architecture's anchor.

The authors have also developed a den-anchor for their proposed YoLo architecture by combining their architecture with two additional models, the denser grid and the anchor box models.

The authors have collected their dataset from various conditions and trained their networks with the following parameters:

- Existing YoLo Architecture:
  - 996 training images containing 1796 potholes and 203 testing images.
  - Learning rate: 1e-5 (from 0 to 100 epochs) and then 1e-6 (from 100 to 200 epochs)
  - Retrained for another 300 epochs
- Their own architecture:
  - 996 training images containing 1796 potholes and 203 testing images.
  - Learning rate: 1e-5 (from 0 to 100 epochs) and then 1e-6 (from 200 to 600 epochs)
  - Trained for another 100 epochs using the Den-anchor

The results were as follows:

TABLE 5. Suone *et. al* method results.

Model	Average Precision	Recall	Parameters	Frame per Second
YoLo v2	60.14%	65.61%	48 million	23
Authors models and anchors	67.74%	74.93%	18 million	32
Authors model and Den-anchor	83.43%	83.72%	18 million	21

Suone *et al.* method did achieve a good experiment result, however, this experiment has a number of limitations:

1- The authors have not provided a real-time testing scenario of their system, results shared where taken directly from the network in offline, non-real-time mode.

2- In real-time scenarios, pothole detection via machine vision using deep learning only isn't feasible due to the fact that many factors could affect the imaging tool, hence putting the service user at risk due to the fact that we will be relying on one system only with no failsafe.

3- The results obtained had low precision (around 82.5%) which cannot be used for safe navigation.

In summary, the assessment as per our criteria is as follows:

1- Efficiency: This system is not very efficient as its accuracy is, according to the authors, 82.43%

2- Real-time functionality: This system can be used in real-time operations but it hasn't been properly tested

3- Computing-power needed: This criterion can be managed as the system can be mounted on an Intel Neural Compute Stick or any other Deep Learning portable small processor.

4- Amount of Power needed: This depends on the processor used and the equipment used.

5- Size of the system: As a deep learning system, this can be manipulated as per the available equipment.

6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case.

b: LASER IMAGING AS INPUT

1. Yu et. al. Method: Yu *et. al.* developed a method in which they have used image processing to extract laser coloured regions in an image.

First, noise has been removed by using a multi-window median filter which uses four filtering masks. Then, the thresholding technique has been used to provide a binary representation of the image, this has been achieved by using Otsu's method [16] which performs automatic thresholding. In order to remove the gaps in the generated binary image, and to connect the close laser line pixels without affecting the area, and to produce smooth boundaries, morphological closing has been used. This method involves dilation succeeded by erosion via the formula $A.B = (A \oplus B) \ominus B A.B = (A \oplus B) \ominus B$ with B being a line structuring element having a size of 20 pixels. The noise reduction has been achieved via the labeling of the components which has been connected and by having the number of pixels which have been connected counted. As a result, pixels within a connected component sharing the same values of intensity have been be interconnected. Based on the connected component's total pixels, any number less than the threshold has been labelled as noise and removed. The system then compares each single frame with the frame considered as a template on a per-tile basis in order to detect any deformations by comparing whether the row having the maximum deformation with the binary value 1 is above or below the row with the maximum deformation with the binary value 0 (if max def. 1 is above max def 0, then the line would be intersecting an obstacle), otherwise the row would be a pothole. This step has been followed by the indexing of the depth and then by the classification of the severity of the distress which has been done by calculating the vertical and horizontal distress, then, along with the depth index and total number of distress tiles, these 4 values have been used in order to make the final decision) which is achieved by using a neural network with the following specifications:

- Number of input nodes: 4
- Number of hidden nodes: 8
- Number of output nodes: 5

This neural network deduces the distress classification as per the author's predefined guidelines.

Their method has been tested using a set of 100 images which includes 10 examples of each distress. The results were not clear as they are simply represented with a table comparing 3 samples (2 potholes and one crack) and showing that the severity level / crack type were the same between the manual assessment and the method proposed.

Many limitations have been identified for this system:

1- The authors did not provide sufficient data in order to assess the system's results
2- The authors did not provide any data in regards to the false positives and false negatives rates in order to assess the performance and reliability of the system
3- The system uses laser imaging which practically cannot be tested with water-filled potholes, no test was also done with such examples
4- The neural network was trained with 100 images only which is not enough for an accurate training.
5- The system does not take into consideration the fact that roads are not flat and it is prone to errors when shaky images are acquired, this was also not reflected in the results mentioned.

The authors did not mention the number of images used for training and for testing

In summary, the assessment as per our criteria has been as follows:

1- Efficiency: This system is not efficient for real-time navigation as the authors did not present any data which can be used in order to assess this criterion.
2- Real-time functionality: Information shared is not sufficient enough to assess this criterion.
3- Computing-power needed: The computing-power needed is significant as this system uses Matlab which requires a large amount of RAM and computing power.
4- Amount of Power needed: Laser imaging equipment and Matlab require a lot of power so this is a limitation when it comes to mounting the system on a moving platform.
5- Size of the system: As a deep learning system, this can be manipulated as per the available equipment.
6- Mass-Production eligibility: This system can not be eligible for mass-production due to the need of using Matlab so a distribution license is required which is not cost-effective, not to forget the need for laser imaging which is also a limitation when it comes to cost management.

c: THERMAL IMAGING AS INPUT

1. Aparna et. al. Method: Aparna *et. al.* have developed a system which detects potholes in real-time via thermal images with the help of a convolutional neural network.

The authors have developed their own convolutional neural network which architecture has been discussed in detail in their paper. In essence, it is a sequential model which takes as input a normalized input which is passed to a series of 2D convolution layers having as activation 3×4 kernel and ReLU. A max pooling layer follows every convolution layer and its output is normalized via batch normalization and passed to a global average pooling. The output has then been passed to a dense layer having a sigmoid activation (classifies the output by either 0 or 1). This model has a loss function which is cross-entropy (logarithmic) and as optimizer, Adam optimizer has been used. As a secondary solution, the authors have also applied transfer learning to different models of ResNet with different images. ResNet has been introduced by [17]. The authors have used many different methods to train the network (cyclic learning rates, differential learning rates, Fastai library on the top of Pytorch in Pythin) in order to achieve their optimal results with an increased accuracy and a reduced overfitting.

Aparna *et. al.* have used a FLIR ONE thermal camera [18] which use "advanced and patented multispectral dynamic (MSX) technology" [19] which provides the merging and extraction of details from the thermal images and the visible images in order to create a series of enhanced images and videos. Their data collection has taken place in Chandigarh city in the northern state of Punjab, India in different times of the day with different lighting and temperature. They have collected images of potholes, water-filled and wet potholes and shades and they obtained a result of 500 images classified with their different criteria (unique identifier for the thermal and vision images, air temp, road temp, pothole temp, time, severity, water-filled or not, shade, and location).

After a series of pre-processing: cropping, resizing, data augmentation, zooming, rotation, mirroring, blurring, enhancing contrast, salt and pepper noise addition images were ready to be used as input to their self-built CNN and ResNet in two different tests.

Their own self-built CNN was used with the following parameters:

- Train-validation split: 90:10
- Image size: 240 × 295
- Total categories: 2.
- Total images: 4904
- Training dataset size: 4320
- Validation dataset size: 480
- Test dataset size: 104
- Kernel: 3 × 3 for convolution layers
- Activation: ReLU for convolution layers
- Loss function: binary cross entropy

As for ResNet, they used 3 different ratios for their training and validation:

- 60:40
- 80:20
- 90:10

They have also used different models of ResNet:

- ResNet18
- ResNet34
- ResNet50
- ResNet101
- ResNet152

The authors have published detailed results of all their test cases in their paper and they were able to achieve the following results:

Their own network:

- Average training accuracy: 55.74%
- Average validation accuracy: 68.99%
- Training and Validation losses on still higher side
- Test accuracy: 73.06%

ResNet:

- ResNet18: Best accuracy 90.52% and validation loss of 27.37%
- ResNet34: Best accuracy 89.42% and validation loss of 27.57%
- ResNet50: Best accuracy 91.77% and validation loss of 24.07%
- ResNet101: Best accuracy 92.50% and validation loss of 22.40%

ResNet152: Best accuracy 91.66% and validation loss of 22.28%

Then, the authors have picked the top 3 performing models (ResNet50, 101 and 152) and have repeated the test with a validation split of 80:20 and two different image sizes which are 224 and 240 (the larger sizes returned an out of memory error). This test has resulted in an increase in accuracy, the results between the two chosen image sizes have been "almost similar" [19]

Finally, they have compared the average results obtained by using their own self-built model and ResNet:

TABLE 6. Using their self-built model.

Avg. Training Accuracy	62.63%
Avg. Validation Accuracy	69.8%

TABLE 7. Using ResNet.

Avg. Training Accuracy	94.64%
Avg. Validation Accuracy	95.2%

They noted their findings stating that using ResNet-based CNN has provided better results than self-built ones and that ResNet50 and ResNet101 have provided the best results which means that they could be used in order to fulfil the desired task. In addition, they have noted that image dimensions 224 × 224 pixels is considered an optimal input dimension for ResNet models and that cyclic learning rates have improved the accuracy by a noticeable amount. Their test has also shown a low rate of false positives.

This system have had a few limitations:

1- The authors combined their test cases in one, the results published do not provide sufficient information regarding the system's performance when used with wet potholes or shiny ones.

2- The Thermal Camera images can be affected by the weather and could provide confused or blurred results.

In summary, the assessment as per our criteria has been as follows:

1- Efficiency: This system is efficient as it provides a high accuracy (more than 95%) and it uses thermal imaging which is very effective in many cases.

2- Real-time functionality: This system can be used in real-time.

3- Computing-power needed: The computing-power needed is significant for training the system but deploying it can be done using any different kind of equipment so this criterion can easily be manageable.

4- Amount of Power needed: This system does not require much power as it requires a controller (can even use

Intel Neural Compute Stick and a Raspberry Pi) and a Thermal Camera which is not very power-consuming.

5- Size of the system: As a deep learning system, this can be manipulated as per the available equipment.

6- Mass-Production eligibility: This system is eligible for mass-production as it relies on a controller and a thermal camera without the need for any expensive proprietary software.

2) NON-DEEP LEARNING TECHNIQUES

a: DETECTION VIA VISIBLE LIGHT RGB CAMERAS

1. Azhar, et al. Method: Azhar, *et al.* have developed a method where they have used supervised learning in order to detect and localize potholes in asphalt pavement images. This technique analyses the surface of the road's features which can be visualized and classifies images as "potholes" or "non-potholes" images including the location at which the pothole is found within the image.

In order to detect potholes, the technique proposes the usage of HOG Feature Extraction: (focused on the shape of object), this method has been based on counting the frequency at which gradient orientation appears in specific parts (portions) of an image.

First, images have been converted to grayscale (from RGB), then, their size has been decreased to 200×200 pixels after normalizing the orientation of the image from 0 degrees to 180 degrees. The image has then been divided into 625 (25×25 pixels) cells which do not overlap. These cells were of size 8×8 pixels each which have been subsequently divided into 4 blocks of size 4×4 pixels each.

Using HOG, a vector of size $1 \times 20,000$ ($625 \times 4 \times 8$) has been obtained. The vector has then been used to classify the image via the Naïve Bayes classifier which labels the input image via the "maximum posteriori probability" technique compute via: $P(Ci \mid Vf) = \frac{P(Vf \mid Ci)\, P(Ci)}{P(Vf)} P(Ci \mid Vf) = \frac{P(Vf \mid Ci)\, P(Ci)}{P(Vf)}$. Ci being the class label with i = 1 and 2. This probability has been used to decide whether the image is a pothole image or not.

The "normalized cuts" technique has then been applied, this technique has been proposed by Chi&Malik [20] which has been used to group perceptual data via the extraction of the globalized "impression" of an image by comparing and measuring the total similarity and dissimilarity between the various groups exiting in an image. [20]

The image has been split into 12 different regions via the "normalized cuts" technique. If any region has been detected to have a threshold of mean having a value which is less than 80, the region would then localized as a pothole.

In order to test the technique, the authors have used a dataset of 120 images gathered by Koch, *et. al.* 50 images were used for training and 70 for testing.

Their results were as follows:

The obtained results have proven that the technique cannot be considered reliable as its accuracy is 90% meaning that there is a 10% margin of error, the precision has also been very low which meant that this technique cannot be used in real-time scenarios as it has a high risk of failure putting the service-user at risk.

TABLE 8. Azhar *et. al.* experiment results.

Accuracy	90.0%
Precision	86.5%
Recall	94.1%

In addition to the previous, other weaknesses have been identifies:

1- The system requires a large amount of calculation, which requires a large computational power, which makes the system incapable of being mounted to a battery-operated platform (the system needed 0.673 seconds in order to detect a pothole in a 200×200 image, nowadays, images are of a minimum of 3456×2304 pixels)

2- The system has a large number of false negatives which means that there are more than one occasion at which the system cannot be relied on for automated navigation purposes.

3- The system relies on HOG features which is known to be prone to many computational errors in the event where the lighting in the image is not enough, or when images are too bright.

4- In addition to the lighting issue, the technique tends to detect shadows as potholes as demonstrated in the paper.

In summary, the assessment as per our criteria is as follows:

1- Efficiency: This system is not efficient for real-time navigation as it has a large number of false negatives and a low accuracy along with low-res images (200×200 px).

2- Real-time functionality: This system cannot be used in real-time operations due to the fact that it can only be used on images.

3- Computing-power needed: This system requires a large amount of computational power as it requires 0.673 secs. To detect a pothole in a 200×200 px image which is a very low-res input.

4- Amount of Power needed: This system requires a large amount of power so it is not compatible with a battery-operated platform.

5- Size of the system: This can be managed as the system requires a processor and a camera.

6- Mass-Production eligibility: This system might be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case.

2. Koch et. al. Method: Koch *et. al* have presented a system which uses segmentation to split images into defective and non-defective using the histogram approach which thresholds based on the shape. They have used "morphological thinning elliptic regression" in order to approximate the potentiality of a pothole being found in a picture using the geometric features of the area flagged as defective, its texture has been compared with the background texture so that the desirable area could be flagged as a pothole. In this method, the authors have replaced a parking camera with a "fish-eye" camera which can be tilted when the car is moving forward in order to collect data which was used in their research. The authors have built their system with 3 main stages, they have started by segmenting the pictures obtained, then, they have extracted the shape and the texture which have been compared with the background.

They have achieved noise-reduction in the segmentation phase by using a 5 × 5 "median filter", then, they have used an algorithm which thresholds data based on the shape using the "triangle algorithm" which have been developed by GW *et. al.* [37]. In order to omit the risk of interference caused by the histogram peaks, the authors have used a "1D median filter" having 5 as its order. The determined threshold T has been used as a value for the intensity of a respective point $P_T = [T, h(T)]P_T = [T, h(T)]$ in the histogram. This point had a maximum distance to the line $I = [P_0, P_{max}]I = [P_0, P_{max}]$ which intersects the origin P_0P_0 of the histogram and $P_{max}P_{max}$ the point which refers to the maximum intensity. The enhanced image $G_{enh}G_{enh}$ has been converted into binary form named B by comparing whether a specific pixel of the enhanced image is less than or equal to the threshold in which case the same pixel in the binary image had 1 as value or 0 otherwise.

The next phase of the method has been the extraction of the shape which has been achieved by removing minimal regions which have a "linear shape" from the binary image along with any other regions which have been, by assumption, not potholes. The shapes from the remaining areas which are potentially potholes have been extracted and three variables have been used: the "major axis" length, the centroid's position and the angel of the orientation. With the use of these variables, areas can be determined as either shade of a pothole or the full pothole.

Then, the "elliptical shape" of the pothole via its shade has been approximated via a series of steps firstly starting by the "morphological thinning" which minimizes the area of the shade in order to obtain the smallest-possible connected "skeleton". Then, the identification of the skeleton's "branching points" took place. "Branching points" have then been connected together so that the "major path" of the shade's area is determined. If the skeleton's end-points were less than 5, the full skeleton would be considered as major path. The approximation of the ellipse have been achieved via the major path elliptic regression method described by Fitzgibbon *et. al.* [38].

The next phase of the method has been the extraction and comparison of the texture which has been achieved after a series a filtering and then by using "morphological dilation" in order to the omit the unwanted result of the filters used, then the binary image has been combined with the inner-region of the defective area and the outside area has been considered the complementary area of the total defective area. Finally, the decision has been made by comparing the vector of features of the candidate areas. If the area is "coarser and grainier", then the area would be considered a pothole.

The dataset used for this method consisted of a total of 120 images where 50 have been used for training and 70 for testing. The system has been manually trained by choosing different values for the thresholds until a maximum performance has been determined.

Their results were as follows:

TABLE 9. Koch *et. al.* test results.

Accuracy	85.9%
Precision	81.6%
Recall	86.1%

This method's limitations are as follows:

1- The accuracy of the system is very low to be used in real-time scenarios.
2- The authors did not mention the runtime required for the system to perform its tasks.
3- The system is prone to errors when it comes to light problems and water-filled potholes
4- The dataset used for training and testing was not made available which makes it hard to validate the test results.
5- The system is based on Matlab and requires a significant amount of computational power which poses a challenge to it being mounted to a moving platform.

In summary, the assessment as per our criteria is as follows:

1- Efficiency: This system is not efficient for real-time navigation as the accuracy is very low (85.9%) and its runtime was not mentioned.
2- Real-time functionality: This system cannot be used in real-time operations due to the fact that it can only be used on images.
3- Computing-power needed: This system requires a significant amount of computational power as it requires the usage of Matlab on a mounted computer.
4- Amount of Power needed: This system requires a large processor if the runtime is to be reduced, in which case the amount of power needed will make it inadequate for battery-operated platforms.
5- Size of the system: This can be manipulated as per the available equipment.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case.

3. Ryu et. al. Method: Ryu. *et. al.* have developed a system inspired from Koch. *et. al* (mentioned previously). Their system detects potholes via data which is collected using an optical device mounted on a moving vehicle and shares the result with different parties (navigation systems companies, Car's On Board Computer (OBC), short-range wireless communication methods etc.).

The system has been formed of 3 phases similar to the Koch. *et. al.* system, "Segmentation", "Candidate Region Extraction" and "Decision" [22]. The first phase would be completed once the regions which are dark have been extracted via the use of the thresholding histogram which uses shape to base its classification, this method together with the maximum entropy method and Otsu's method [16]. Then, noise has been removed via the use of a "median filter". Ryu. *et. al.* have tested 3 sizes of the filter (3×3, 7×7 and 9×9) and have chosen 9×9 based on its performance, then, they have restored the outlines which have been damaged in the regions of the object and remove the minimal pieces via the "closing operation (dilation and erosion)" with the use of a 7×7 square morph filter (Morphology), then, candidates have been extracted via different features ex. how compact they have been and their size using a series of formulas followed by a modification of the previously mentioned histogram method which ensures the separation of the pothole and the region which have been bright (road lane marks etc.). The last phase has been achieved via a comparison between the features of the background and the candidate pothole itself which have resulted to a decision of whether the candidate is a pothole or not via the OHI or Ordered Histogram Inspection method proposed by Van Der Weken *et. al* [36] which have been used to discern light, stain, patch etc. This has been achieved by using the size of the defined region, its compactness has been calculated using the formula $com\left(M_c'\right) = \frac{l^2}{4\pi A} com\left(M_c'\right) = \frac{l^2}{4\pi A}$ with ll being the perimeter and A the area of the region. The method has also used the refined version of the region which is a candidate which contains features such as the "compactness, center point, and convex hull" which refines incomplete candidate regions via the OHI method which measures how similar are the different regions of the image. The region would be considered a non-pothole region if the standard deviation of the refined candidate region has been less than threshold of the of the standard deviation, or if:

OHI of the refined candidate region and background region is larger than the threshold of the OHI, or if the OHI of the refined candidate region and background region is larger than the threshold of OHI values calculated by Sobel Operation of if the out region's standard deviation negated by the inner region's standard deviation was less than the threshold of the standard deviation obtained via Sobel Operator of if the out region's average negated by the inner region's average is larger than the threshold of the average outer region being the outside of the candidate region which was refined and inner region being the inside of the candidate region which was refined.

Otherwise, this region would be considered a pothole region.

In their testing phase, the authors have collected a total of 90 images split into 20 images of asphalt roads, 20 concrete roads which were chosen randomly, along with 10 original images and 10 close-up ones, and 10 bright images and 10 dark ones. The average time needed for their system to produce a result has been 46.8 seconds with a maximum of 218.2 sec. and a minimum of 13.7 sec.

This method has been compared with Koch *et. al* (mentioned in the previous section) and according to the authors, it has achieved a higher performance.

Their results were as follows:

TABLE 10. **Ryu *et. al.* test results.**

Accuracy	73.5%
Precision	80.0%
Recall	72.3%

This result has proven that the technique cannot be considered in real-time operations for critical systems as there has been a considerable probability of risk not to forget the amount of time required for the system to produce values as the minimum processing time has been 13.7 sec. which is not enough for real-time operations.

Additionally, other weaknesses and issues have been identified:

1- This system is weak when it comes to light, including shadows, as the authors have mentioned that false detection occurs depending on the type of the shadow found in the picture.
2- The large number of false-negatives is an important limitation to the usage of this system.
3- The system is prone to error when the vehicle is not stable, which cannot be expected when navigating in real life using a wheelchair.
4- The amount of processing time and power required makes this method a bad choice for real-time navigation.
5- The dataset used to test this system was not made available which prevents the validation of the test results.
6- The results published in this paper are significantly less than the Koch *et. al.* results even though the author mentioned that their results are better, their version of the Koch. et. al performance shows 45.1% accuracy while Koch *et. al.*'s mentioned performance is 85.9% (as mentioned in the previous section) which could be linked to the nature of the dataset used and the difference in the testing data.
7- This system was not tested with potholes filled with water nor with over-illuminated potholes.

In summary, the assessment as per our criteria has been as follows:

1- Efficiency: This system is not efficient for real-time navigation as the input is of size 200×200 px and the accuracy is very low (73.5%).
2- Real-time functionality: This system cannot be used in real-time operations due to the fact that it can only be used on images and requires a long time (minimum of 13.7 sec.) to produce an output.

3- Computing-power needed: This system requires a significant amount of computational power as it requires a minimum of 13.7 sec. and an average of 46.8 sec. to complete the task.
4- Amount of Power needed: This system requires a large processor if the runtime is to be reduced, in which case the amount of power needed will make it inadequate for battery-operated platforms.
5- Size of the system: This can be manipulated as per the available equipment
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res camera which could be a deal-breaker in this case

4. Schiopu et. al. Method: Schiopu *et. al.* have defined a method which uses video sequences taken by normal cameras as input in order to detect and track potholes after using its threshold algorithm to create the set of candidates within the area being considered. Their method relies on the concept that the representation of potholes in the intensity images is via high values. They have used an algorithm which relies on thresholding in order to generate the region of candidates set via the selection of areas of the image with the highest intensity values. The system then distinguishes shadows from real potholes via the criteria: size of pixel, regularity, depth via estimation, shape and length of the contour, and whether the pothole appears in consecutive frames. The system's first step has been the selection of the region of interest (ROI). This procedure has been offline and have followed the concept of the road having two lines which are in parallel intersecting at a point names the vanishing point (V), and that the search area has been between the hood of the car up to a distance where smallest potholes can still be seen. The thresholding algorithm has then used as a method for the removal of the pixels which represent the wayside. This algorithm has calculated the intensity image at the studied frame and has searched for the pixels within the ROI having the intensity less than the calculated threshold, i.e. the darkest pixels. The system has then removed object reflections via an offline procedure applied to consecutive frames. This procedure has used mean intensity image and depth matrix and has compared the depth matrix with the standard deviation of the depth values, this way it would detect the reflections and removes them. This step has been followed by the shadow detection which relies on many different properties such as: the region's model: the regularity of the shape of the reflection of an object in oppose to the random shape of the pothole, the depth of the pothole detected via the number of dark pixels within the candidate area, the length of the contour via the boundary pixels, the shape of the contour as it would be more straight in the case of a reflection rather than the case of a pothole. Then, the system labels potholes if the candidate region passes the entire process without being filtered out. The system then focuses on the consecutiveness of the frames in order to keep a live track of the pothole where it detects the new position by calculating the previous position and negating it from the speed of the car. This process uses the Euclidian distance and relies on many variables such as the estimated car speed, the height of the camera placement, the camera's angel of view and keeps track of the pothole.

The authors have carried their testing using Samsung Galaxy S4's front camera and drove for 34 minutes on a road in Hervanta, Kalvola. The resolution used has been 1080×1920 at 30 frames per second with dry conditions and a clear sky.

Their algorithm has been implemented via Matlab and they were able to extract 61200 frames in order to be tested and were able to achieve 55 detections with 6 false positives and 0 false negatives with a runtime of 27.24 seconds to check and detect potholes within 639.90 seconds of data collection.

Their results were as follows:

TABLE 11. Schipou *et. al.* test results.

Precision	90.0%
Recall	100%

This method had a number of weaknesses which have been identified:

1- This method has a precision of 90% which is good but not good enough for a critical system such as a moving platform with real users.
2- This method estimates the speed and calculates its tracking variables based on the estimation made which could be problematic at different speeds.
3- This method does not account for inconsistency of frames in the event where a strong light is subjected to the camera.
4- This method was not tested during the night in order to test its performance as the number of dark pixels will be really challenging.

In summary, the assessment as per our criteria is has been as follows:

1- Efficiency: This system can be efficient with some slight modifications in order to raise its precision.
2- Real-time functionality: This system can be used in real-time as it relies on videos so it can be modified to run on a real-time stream, it requires further testing in order to decide whether its detection is achieved in real-time or if it requires more time.
3- Computing-power needed: This system's computing power is acceptable (27.24 seconds within 639.9 seconds of runtime) but it requires further testing in order to identify computing power used (resources).
4- Amount of Power needed: This system can be considered for low power consumption as it only relies on a camera and a processor.
5- Size of the system: This can be manipulated as per the available equipment.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the

equipment needed, the only limitation for this criterion would be the need to use Matlab so unless their code can be mounted to different platform via Matlab's toolkits, it will pose an issue.

5. Dihao et. al Method: Dihao *et. al* have developed a method which uses the Probabilistic Generative Model (PGM) in order to calculate the probability of an occurrence of a pixel representing a crack via the intensity details and information. The system then illustrates and analyzes the capability for detection using the information acquired. They have also used the Support Vector Machine (SVM) approach in order to obtain the probability of a pixel representing a crack using the information obtained from the neighborhood. The system then compares the capability of detection via the info obtained from various neighborhoods scale. In order to improve the accuracy, their system fuses the different probabilities of each pixel and uses weighted dilation as a method to improve the detection and recognition of the pixels of the borderline along with the continuity of a crack without any increase in the crack's widths. The first step in their procedure is to create a probability map which bases on the intensity of the pixel. This is achieved via the computation of the posterior distribution of pixel intensities. They tested both the PGM model and the threshold-OTSU [16]. The authors have compared the results obtained via both methods and concluded that the PGM model has a better performance in comparison to the OTSU method's performance when it comes to intensity pixels. The next step is to create a probability map based on the information obtain from the neighborhood pixels for which the authors have calculated a probability vector holding the probability that a pixel is considered as a pixel representing a crack based on the information obtained from its neighborhood and the size of the neighborhood. They have defined their own method for the computation of the probability vector (the method is extensively explained in their paper).

Then, the authors have introduced a method for data fusion which combines the probability obtained via pixel intensity and the probability vector calculated based on the information obtained via the neighborhood pixels. Together, both probabilities fused provide a higher detection accuracy rate. This fusion algorithm uses the mean of the probability vector along with a series of instructions which include a max operation and a multiply operation. The algorithm can be found in their paper along with its explanation. The final step of the Dihao *et. al* method is the weighted dilation which is used because of a constraint which was introduced by the authors in the previous step. This constraint is that pixels with high probability are considered cracks and this method neglects the border pixels of the crack which cannot hold a high probability. Their improved dilation algorithm has been called weighted-dilation and it uses the probability maps which were previously obtained. The weighted-dilation consists of computing the structing element's mean probability and comparing it with a weight defined as 0.5 (their decision condition).

This method has been implemented in Python and the testing parameters were as follows:

- Dataset: CFD [23] consisting of 118 images (with noisy pixels such as spots with oils and stains of water and bad light conditions)
- Number of images used for training: 70
- Number of images used for testing: 48

The authors have tested their method with 3 tolerance margins (0, 1, and 2) and their best results were obtained with the 2-pixels margin:

TABLE 12. Dihao *et. al.* test results.

Precision	90.7%
Recall	84.6%
F1-Score	87.0%

This method has a number of weaknesses which were identified:

1- This method has a precision of 90% which is good but not good enough for a critical system such as a moving platform with real users.
2- This method requires a large number of heavy computations which can be problematic when it comes to real-time functionality and the amount of power required.
3- The authors did not present enough information in regards to their testing with water-filled cracks and other noisy images.
4- This method was not tested during the night in order to test its performance as the number of dark pixels will be really challenging.
5- The authors did not provide any information in regards to the runtime of the system and the amount of time it needs in order to detect the crack.

In summary, the assessment as per our criteria has been as follows:

1- Efficiency: This system is not efficient enough to be used in real-life for a critical system.
2- Real-time functionality: This system cannot be used in real-time as it requires a large amount of computational power which could be an issue, not to forget that the authors did not provide sufficient information in regards to the runtime of the system.
3- Computing-power needed: This system requires a large amount of computational power.
4- Amount of Power needed: This system requires a large amount of power.
5- Size of the system: This can be manipulated as per the available equipment.
6- Mass-Production eligibility: This system is eligible for mass-production but might have a high cost due to the need of a powerful processor in order to achieve the requires computational task.

b: DETECTION VIA STEREO VISION

1. He Youquan, et al. Method: He Youquan *et. al.* have developed a system which detects potholes via the concept of three-dimensional projection transformation which produces the pictorial details related to the pothole. This method fuses a number of different techniques in order to preprocess images and to analyze the results, these methods include binarization, thinning, reconstructing images in three-dimensional space and the analysis of errors along with their compensation.

The system has been calibrated in such a way that the physical location of the section of the area due to be tested is determined. Binocular stereovision was used in order to achieve this task by using two cameras in order to achieve the localization of the position. After calibration, a simplistic method has been used in order to provide coordinates for the image after binarization. This step is followed by another step where the image side coordinates are transformed into the object's side coordinates via the combination of the RAC two-step method and the orthogonal least-squares method and by using the obtained matrices values in a conversion formula presented in the paper. The result is a determined relation between the object side coordinates and the image side coordinates which concludes the calibration phase. This is followed by the denoising of the image by using the "neighborhood averaging" [24] method which filters and removes any mutated pixels. This technique is based on the Otsu Thresholding technique [16] which results into the removal of the background and the appearance of a band of light which is clear. Then, the light band's colour is inverted, expanded and then contracted in order to remove the noise points which are isolated. Faultage is then reduced and the originally thin light band picture becomes a curve with a width of one pixel. With the help of the previously obtained coordinates relationship, the object side coordinates are calculated.

According to the authors, this system has produced a result with 2 mm discrepancy which has been the only result shared.

This method has had some significant weaknesses:

1- This system can perform detection but not localization.
2- Intensity of the LED light affects the results very badly as it can affect the image obtained by the CCD Camera.
3- The CCD camera's resolution and performance has a direct impact on the result.
4- External light can cause a direct interference factor which affects the results significantly.
5- The paper does not provide enough data in regards to the results obtained and the accuracy of the system.
6- The paper does not provide any data in regards to the environment (location, time of the day, weather conditions) in which the system has been tested.
7- No error rate data has been provided either.

In summary, the assessment as per our criteria has been as follows:

1) Efficiency: Not enough evidence has been provided in order to assess this criterion.
2) Real-time functionality: Not enough evidence as the runtime of the system has not been provided, also, this system only detects potholes but does not localize them, and needs to be on the top of the pothole at a short distance.
3) Computing-power needed: From the system's description, the computing power required is manageable on a moving platform.
4) Amount of Power needed: This system can be mounted to a moving platform's battery.
5) Size of the system: This system's size is acceptable as it relies on an LED light, CCD Camera and a processor.
6) Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed, the only limitation for this criterion would be the need of a high-res CCD camera which could be a deal-breaker in this case.

2. Zhang, et al. Method: Zhang. *et al.* have proposed a real-time method which detects potholes via the use of stereo vision.

Their system consists of a number of steps, the first being a disparity calculation algorithm which they have proposed in a different paper [REFEREMCE NUMBER]. Their algorithm calculates the disparity range for a certain pixel in a specific image line by using the disparity values obtained at three pixels which are neighbours to the direct lower image line. Their disparity calculation algorithm is formed of 3 steps ("cost computation" matching, "controlled search range" and "disparity enhancement") [25].

This step has been followed by the detection of the pothole where they use a surface-fitting algorithm in order to estimate the surface of the road so that any points which are less than the main surface of the road are considered potholes.

The first part of the pothole detection is a conversion of the points to the Euclidean space (originally being in the disparity space) which is achieved by the use of a simple formula which relates the calibrated coordinates to the focal length, the formula has been sufficiently explained in the paper.

The next step is the fitting of the surface which was achieved via a "low computational bi-square weighted robust least-squares method" [26] described in papers [27] and [28].

Then, in order to define the model of the road, they have introduced the equation $z = a_1 + a_2x + a_3y + a_4x^2 + a_5xy + a_6y^2z = a_1 + a_2x + a_3y + a_4x^2 + a_5xy + a_6y^2$ which takes into account the twists and bends of the surface of the road.

The method uses this formula which defines the true surface and subtracts it from the surface which was estimated. The pothole is detected when this subtraction is larger than a certain defined threshold which they set to 0.04m in their test. The method creates a segmentation of the disparity image and uses the "connected component labelling algorithm" [26] in order to label the potholes.

The authors have represented their results via 4 examples showing the region of interest and the surface fitting results represented via a graph. The results have shown some examples of false positives and false negatives and a shared short video of their live test.

This method has been optimized by [29] where they have made the method real-time by making some improvement to the algorithm. They have removed the V-disparity step which has been used as a noise filtering technique. They have also sampled the data during curve-fitting using the RANSAC (Random Sample Consensus) algorithm. In addition, they have made some optimization steps such as memory management, down sampling of the region of interest and decreasing the reliance on math formulas in any library which has been external and have replaced it with lookup tables. They have even eliminated if statements and unrolled the loops in order to achieve an optimized result. This has decreased their runtime by a significant amount of time. They have also used the parallelisation of the code technique in order to improve performance by using a parallel code split on more than one core of the processor using OpenMP API in order to ensure a proper threading. Mikhailiuk *et. al* have stated that they tested their improved system on 3000 images and were able to achieve 145 frames per second and the following:

TABLE 13. Mikhailiuk *et. al.* test results (improved Zhang. *et. al* method).

Accuracy	98.0%
Recall	100%

This method (including its modified version) has had some weaknesses:

1- This system's results are affected by the light intensity so the system can be faulty when the light is weak or too strong.
2- The system will fail when trying to detect water or ice-filled potholes as the cameras will not be able to produce a valid output.
3- The paper does not provide enough data in regards to the results obtained and the accuracy of the system.
4- The paper does not provide any data in regards to the environment (location, time of the day, weather conditions) in which the system has been tested.
5- No error rate data has been provided either.

In summary, the assessment as per our criteria for both Zhang. *et. al* and Mikhailiuk *et. al's* versions has been as follows:

1- Efficiency: Not enough evidence has been provided in order to assess this criterion.
2- Real-time functionality: This system and its improved version can both be used in real-time.
3- Computing-power needed: Both systems' computing power required is manageable on a moving platform.
4- Amount of Power needed: This system can be mounted to a moving platform's battery.
5- Size of the system: This system's size is acceptable as it relies on a stereo imaging camera along with the required processor.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed.

3. Li, et al. Method: Li *et. al* introduced a method that used stereo vision in order to provide 3D measurements which will be used to extract the pothole's geometry features. This method is split into two steps, the offline step and the online step.

In the offline step, the parameters (intrinsic and extrinsic ones) are obtained from the camera via the use of a checkerboard which uses Zhang's calibration method mentioned in this paper. The authors have used a "8 × 6 checkerboard and 24.5 mm squares" in order to calibrate their stereo camera.

The online step consists of three modules, processing the images, calculating the disparity and detecting the pothole. These steps are prerequisites before transferring the coordinates from the images to the real world. After obtaining an image through the calibrated depth camera, the system calculates the disparity via an algorithm explained in the paper which relies on the left and right optical centers, baseline and the real point in the real world. Having generated the disparity map, the authors triangulated the results so that the disparity image is re-projected onto the 3D space, this allowed the authors to calculate the 3D coordinates of every point. After obtaining the coordinates, the system fits the road surface and regards all points as equal quality. The points below the road surface will be the potholes. The authors have used the bi-square weighted robust least-squares method [26] in order to "minimize the outliers' influences" when fitting the surface. This is done via adding the weight as an extra factor for scaling. Outliers which are below the surface will be considered as pothole regions.

The authors have used the connected component labelling algorithm [57] which relies on two passes in order to label the regions of interest as potholes and provides the final result of the detection algorithm.

The judgement process is explained extensively in the paper.

In order to test the algorithm, the authors have used two USB cameras which they mounted over a roller cart. They have chosen Raspberry Pi 2 model B as a processor running OpenCV and a Python code. The images obtained were of size 640 × 480 and the time needed in order to detect a pothole was around 4.94s.

The authors did not provide any data which relates to the success and failure rates of the system which makes assessing the performance of the system hard.

This method has had some noticeable weaknesses:

1- The system's performance relies on two stereo cameras, which need to be calibrated every time the code runs.
2- The system will fail when trying to detect water or ice-filled potholes as the cameras will not be able to produce a valid output.
3- The system's performance will be severely affected by the light intensity as it relies solely on two RGB cameras.
4- The paper did not provide any data which can be used to assess the success and failure rates of the system.

5- The paper only mentions the location where the system is tested but does not contain any data which describes the time and weather conditions in which the system was tested.

In summary, as per our criteria, the system's assessment is as follows:

1- Efficiency: Not enough evidence has been provided in order to assess this criterion.
2- Real-time functionality: This system can be used in real-time.
3- Computing-power needed: The systems' computing power required is manageable on a moving platform.
4- Amount of Power needed: This system can be mounted to a moving platform's battery.
5- Size of the system: This system's size is acceptable as it relies on a stereo imaging camera along with the required processor.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed.

c: DETECTION VIA DEPTH CAMERA

1. Moazzam et. al. Method: Moazzam *et. al.* have developed a method which relies on a depth camera (Microsoft Kinect Sensor) explained in references [30]–[32] in order to detect potholes and provide information in regards to its area, depth, length, width and volume.

Their system consists of a Kinect Sensor located between 0.8 and 0.9 meters above the ground. This sensor's data is accessed through Matlab where the processing occurs. Then, it detects the pothole which is the local minimum in every column subtracted from the column itself. The Z-axis being in real-world coordinate system represented in millimeters, the x and y coordinates are calculated as they are represented in pixels. The authors have used a formula which relates the real-world coordinate system to the normalized coordinates value and a constant value which relates to the Kinect's field of view. Then, the mean, maximum depth and standard deviation of the detected potholes have been obtained via Matlab's built-in functions. The area is obtained via the conversion of depth images to binary for "every millimeter increment in depth" [33]. This value has been obtained in pixels and is transformed to real-world coordinates via the multiplication with the area of one pixel in world-coordinates at the particular depth. In order to approximate the volume of the pothole, the system plots the area vs depth curve and uses the trapezoidal rule with unit spacing. It also generates contour plots in order to display the depth slicing (every colour represents one depth level in mm). The authors have also provided a 3D plot if a certain pothole in real-world coordinates in mm in order to represent their results. The authors have chosen three classification types for potholes: Squared Decay, Longitudinal and Cube-like. This classification is based on the area decrease and decay. This comes with many different geometrical values which the algorithm calculates such as centroid, eccentricity, orientation.

This method's weaknesses have been the following:

1- This system's results are affected by the light intensity so the system can be faulty when the light is weak or too strong.
2- The system will fail when trying to detect water or ice-filled potholes as the cameras will not be able to produce a valid output.
3- The error rate of the system is around 15% which is high.

In summary, the assessment as per our criteria has been as follows:

1- Efficiency: This system can be considered partly-efficient as it provides many different calculations and measurements but its error-rate is around 15% so it needs to be improved.
2- Real-time functionality: Not enough data has been provided in order to assess this criterion (runtime was not mentioned)
3- Computing-power needed: From the system's description, the computing power required is manageable on a moving platform, the only issue is with the use of Matlab.
4- Amount of Power needed: This system can be mounted to a moving platform's battery.
5- Size of the system: This system's size is acceptable as it relies on a depth camera along with the required processor.
6- Mass-Production eligibility: This system can be eligible for mass-production due to the nature of the equipment needed if Matlab can be replaced with an open-source system or a different programming language.

d: DETECTION VIA TIRE PRESSURE/VIBRATION

1. Use of Accelerometers:

[42]–[44] have introduced a method which uses smartphone built-in accelerometer, compass and GPS in order to detect potholes. Others such as [60] have used ultrasonic sensors along with accelerometers which are fixed on a flat surface in conjunction with Arduino Uno and ESP8366 in order to measure the depth of the pothole and to detect it along with its GPS coordinates. These methods have relied on the fact that a pothole causes a change in the vertical line detected via an accelerometer as the pothole would cause a noticeable vibration. On this basis, the tire's sound, pressure and vibration can be detected and used in order to flag potholes. These algorithms have been later on improved and replaced with other systems which rely on the a detection based on the car's axle in order to get the best possible results. Such systems have been developed by [45] and [46] all of the previous methods calculate the IRI or International Rough Index. Other variations of the concept have been introduced by [47], (Zhang *et. al* 2013), [48], [53] who have detected the acoustic noise generated by the tire when being subjected to a pothole while [53] and [49] have relied on pressure sensors in order to detect potholes via the change in tire

pressure generated when the tire hits a pothole. [50] have even introduced a deep learning approach in order to fulfil this task but due to the fact that this method relies on vibrations, it has not been added to the deep-learning methods.

All these methods cannot be used in real-time detection for autonomous vehicles as they are methods which can only be used to detect a pothole after being subjected to it, i.e. they cannot be used in pothole/crack avoidance as they simply detect where the pothole is after hitting it.

Many weaknesses have been determined for these systems:

1- These systems, as mentioned, rely on post-detection of the pothole so they cannot be used for negative surface avoidance.
2- These systems rely on tire pressure or vibrations which can be caused by many different factors such as uneven roads, curbs, etc.
3- These systems do not provide any localization method in order to perform avoidance.
4- These systems are prone to many different outside factors.

This type of systems could not be assessed through our criteria as it simply cannot be used for pre-detection and avoidance.

These systems have not been included in the assessment tables as they could only detect potholes after they are traversed by the tire of the automated platform, which meant that a user has to actually traverse a pothole in order to have it detected by the system. Event sensors such as the ones used in these methods could not be used in order to achieve the task proposed hence they have not been mentioned and the only techniques to be mentioned are the forthcoming sensing techniques (pre-detection).

III. ASSESSMENT OF THE EXISTING TECHNIQUES

In order to illustrate the assessment of the techniques described, Tables 14 and 15 have been made available, the techniques have been split between vision and non-vision techniques.

It can be observed that most techniques for the detection of negative road anomalies have been oriented towards the use of computer-vision due to the randomness and stochasticity of the nature and different features of a negative surface, such as location, shape, colour, form, depth, and many others. Most of these vision techniques have failed to produce a good result in strong lighting, low lighting and water or ice-filled potholes or cracks, this would pose a real threat to the service user because these cases exist in everyday life and should be managed. On the other hand, other techniques such as Laser or Thermal imaging have tended to provide promising results but also have their own limitation in many cases ranging from reflection of the laser beam in the case of laser imaging to the outside temperature which could have caused a significant amount of noise in the case of thermal imaging.

The existing techniques for negative road anomalies detection which have been discussed in this review paper have not been sufficient for the fulfilment of the guidance task for automated vehicles and moving platform due to their weaknesses which, in many occasions, could cause different risks to the safety of the service user. Most of these techniques could have been used for a simple detection task where reliability and fault-tolerance are not an issue but when it comes to critical systems, these methods cannot be used. Some of these methods do provide real good results but with minor or moderate weaknesses which could have been caused by many different factors such as the nature of the equipment used, the data-acquisition equipment which in most of the cases has had its own limitation which affects the output of the system. The computation power and real-time functionality are also an issue in many cases along with numerous different factors. The limitation of the acquisition system mentioned earlier poses a high risk which leads to making the system unusable considering the acquisition technology made available at the time when this review paper has been written. This has raised a need for a new technique/algorithm which combines the results obtained from different systems and acquisition techniques making sure that the techniques chosen to complete each other's weaknesses and limitation in order to produce the most optimal possible solution. This way, the result would have been a system able to consider many different factors obtained from many different sensors and techniques and make a decision based on the various input steams provided.

IV. CONCLUSION

Current research has been extensively focusing on machine vision more than the other techniques. Every technique has had its own limitation and weaknesses which could cause a significant risk to the service users hence making these techniques not usable for real-time navigation of autonomous vehicles and platforms. This limitation has long restrained the capability of such vehicles. Finding a complete system which provides autonomous avoidance of negative obstacles to autonomous vehicles has always been a challenging task due to the stochastic nature of pavements and footpaths, potholes and cracks exist in different shapes, and could be filled with water, ice, dirt, or could be reflecting a strong light etc. every case is a limitation to a certain detection system ranging from RGB cameras where water/ice, low light and strong light are a limitation, to thermal cameras where high temperature is a limitation, to reflective laser, where reflection caused by water/ice is a limitation. Not to forget the limitation in the processing technique or power needed, as some systems require a heavy amount of computation, while others require a large amount of power in order to power the sensor/processor. An additional issue is the real-time functionality as not all systems can be used in real-time and this task should be fulfilled in real-time with a very low runtime as the detection should be as close to instant as possible in order to provide an accurate avoidance. Finally, the size of the system could be in some cases a limitation as some systems require larger equipment or larger power source which could not be mounted onto the autonomous vehicle. This could be managed in most of the cases but it has to be considered as an important factor

TABLE 14. Vision systems comparison table.

Authors	Method	Performance				Is Deep Learning?	Main Limitation	Efficiency	Real-time	Computational Power	Power Needed	Size	Mass-production
		Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)								
Pereira et al.	Camera + CNN	98.8	100	99.6	99.6	Yes	Can only be used on images	Efficient	No	Low for low-res images, manageable	Manageable	Manageable	Yes
Anand et. al. method	Camera + CNN	99.893	NM	NM	72.14	Yes	Texture-based	Partly-efficient	Yes	Manageable	Manageable	Manageable	Yes
Gopalakrishnan et. al.	Camera + DCNN	90.0	90.0	90.0	90.0	Yes	Lack of proper testing	Partly-efficient	Yes	Manageable	Manageable	Manageable	Yes
Azhar et. al	Camera + Supervised Learning	90.0	86.5	94.1	NM	No	Large amount of calculations	Not Efficient	No	Large amount	Manageable	Manageable	Yes
Koch et. al	Camera + Segmentation	85.9	81.6	86.1	NM	No	Low performance	Not Efficient	No	Large amount	Large amount	Manageable	Yes
Ryu et. al	Camera + Segmentation (improved Koch. et. al)	73.5	80.0	72.3	NM	No	Low Performance	Not Efficient	No	Large amount	Large amount	Manageable	Yes
Schiopu et. al	Camera + Thresholding	NM	90.0	90.0	NM	No	Different speeds of the platform	Acceptable	Yes	Acceptable	Manageable	Manageable	Yes
Dihao et. al	Camera + PGM + SVM	NM	90.7	84.6	87.0	No	Large computational power	No	No	Large amount	Large amount	Manageable	Yes but high cost
He Youquan et. al	Stereo Vision	NM	NM	NM	NM	No	Intensity of light	NED	NED	Manageable	Manageable	Manageable	Yes
Zhang et. al. + Mikhailiuk et. al (Improved)	Stereo Vision	98.0	NM	100	NM	No	Light Intensity + Water-filled potholes	Yes	Yes	Manageable	Manageable	Manageable	Yes
Moazzam et. al	Depth Camera	NM	NM	NM	NM	No	Intensity of light	Partly-efficient	NED	Manageable	Manageable	Manageable	Yes
Suone et. al	Camera	NM	82.43	83.72	NM	Yes	Low Performance	Not Efficient	Yes (not tested)	Manageable	Manageable	Manageable	Yes
Li et. al	Depth Camera	NM	NM	NM	NM	Yes	Light Intensity + Water-filled potholes	NED	Yes	Manageable	Manageable	Manageable	Yes

NM: Not Mentioned
NR: Not Relevant
NEE: Not Enough Data

TABLE 15. Non-vision systems comparison table.

Authors	Method	Performance				Is Deep Learning?	Main Limitation	Efficiency	Real-time	Computational Power	Power Needed	Size	Mass-production
		Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)								
Yu et al.	Laser Imaging	NM	NM	NM	NM	Yes	Water-filled potholes	NED	NED	High	High power	Manageable	No
Aparna et. al	Thermal Imaging	95.2	NM	NM	NM	Yes	Not enough test information	Efficient	Yes	Manageable	Low	Manageable	Yes

NM: Not Mentioned
NR: Not Relevant
NEE: Not Enough Data

for this task. One additional limitation could be the ability to mass-produce the system which in most cases could be managed but the cost of the system might be high for the users.

REFERENCES

[1] F. Chollet, *Deep Learning With Python*. New York, NY, USA: Manning, 2017.

[2] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," 2015, *arXiv:1511.02680*. [Online]. Available: https://arxiv.org/abs/1511.02680

[3] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[4] V. Badrinarayanan, A. Handa, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," 2015, *arXiv:1505.07293*. [Online]. Available: https://arxiv.org/abs/1505.07293

[5] F. N. Iandola, S. Han, M. W. Moskewicz, and K. Ashraf, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*. [Online]. Available: https://arxiv.org/abs/1602.07360

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1–9.

[7] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3708–3712.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," presented at the 3rd Int. Conf. Learn. Represent. (ICLR), San Diego, CA, USA, 2014.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 248–255.

[10] P. Cunningham and S. Delany, "K-nearest neighbour classifiers," Tech. Rep. UCD-CSI-2007-4, Mar. 2007.

[11] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, pp. 5–32, Oct. 2001.

[12] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, pp. 3–42, Mar. 2006.

[13] T. Evgeniou and M. Pontil, *Support Vector Machines: Theory and Applications*. Berlin, Germany: Springer, 2001. [Online]. Available: https://link.springer.com/chapter/10.1007/3-540-44673-7_12

[14] Deeplearning.Stanford.Edu. (2019). *Unsupervised Feature Learning and Deep Learning Tutorial*. Accessed: May 30, 2019. [Online]. Available: http://deeplearning.stanford.edu/tutorial/supervised/LogisticRegression/

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, and M. E. P. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[16] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[18] FLIRONE. (2019). *FLIR ONE Gen 3 Thermal Camera for Smart Phones | FLIR Systems*. Accessed: May 28, 2019. [Online]. Available: https://www.flir.co.uk/products/flir-one-gen-3/

[19] Aparna, Y. Bhatia, R. Rai, V. Gupta, N. Aggarwal, and A. Akula, "Convolutional neural networks based potholes detection using thermal imaging," *J. King Saud Univ.-Comput. Inf. Sci.*, Feb. 2019, doi: 10.1016/j.jksuci.2019.02.004.

[20] J. Chi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[21] K. Azhar, F. Murtaza, M. H. Yousaf, and H. A. Habib, "Computer vision based detection and localization of potholes in asphalt pavement images," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2016, pp. 1–5.

[22] S.-K. Ryu, T. Kim and Y.-R. Kim, "Image-based pothole detection system for ITS service and road management system," *Math. Problems Eng.*, vol. 2015, Sep. 2015, Art. no. 968361, doi: 10.1155/2015/968361.

[23] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 12, pp. 3434–3445, Dec. 2016.

[24] H. Youquan, W. Jian, Q. Hanxing, W. Zhang, and X. Jianfang, "A research of pavement potholes detection based on three-dimensional projection transformation," in *Proc. 4th Int. Congr. Image Signal Process.*, Oct. 2011, pp. 1805–1808.

[25] Z. Zhang, X. Ai, and N. Dahnoun, "Efficient disparity calculation based on stereo vision with ground obstacle assumption," in *Proc. 21st Eur. Signal Process. Conf. (EUSIPCO)*. Marrakesh, Morocco: European Association for Signal Processing, 2013.

[26] 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.

[27] G. H. Golub and U. von Matt, "Quadratically constrained least squares and quadratic problems," *Numerische Math.*, vol. 59, no. 1, pp. 561–580, Dec. 1991.

[28] X. Ai, Y. Gao, J. G. Rarity, and N. Dahnoun, "Obstacle detection using U-disparity on quadratic road surfaces," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1352–1357.

[29] A. Mikhailiuk and N. Dahnoun, "Real-time pothole detection on TMS320C6678 DSP," in *Proc. IEEE Int. Conf. Imag. Syst. Techn. (IST)*, Chania, Greece, Oct. 2016, pp. 123–128.

[30] R. A. El-laithy, J. Huang, and M. Yeh, "Study on the use of microsoft kinect for robotics applications," in *Proc. IEEE/ION Position, Location Navigat. Symp.*, Myrtle Beach, SC, USA, Apr. 2012, pp. 1280–1288.

[31] D. S. O. Correa, D. F. Sciotti, M. G. Prado, D. O. Sales, D. F. Wolf, and F. S. Osorio, "Mobile robots navigation in indoor environments using kinect sensor," in *Proc. 2nd Brazilian Conf. Crit. Embedded Syst.*, Campinas, Brazil, May 2012, pp. 36–41.

[32] P. Rakprayoon, M. Ruchanurucks, and A. Coundoul, "Kinect-based obstacle detection for manipulator," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*, Kyoto, Japan, Dec. 2011, pp. 68–73.

[33] I. Moazzam, K. Kamal, S. Mathavan, S. Usman, and M. Rahman, "Metrology and visualization of potholes using the microsoft kinect sensor," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1284–1291.

[34] V. Pereira, S. Tamura, S. Hayamizu, and H. Fukai, "A deep learning-based approach for road pothole detection in timor leste," in *Proc. IEEE Int. Conf. Service Oper. Logistics, Inf. (SOLI)*, Jul. 2018, pp. 279–284.

[35] C. Koch and I. Brilakis, "Pothole detection in asphalt pavement images," *Adv. Eng. Inform.*, vol. 25, no. 3, pp. 507–515, Aug. 2011.

[36] D. Van der Weken, M. Nachtegael, and E. Kerre, "Some new similarity measures for histograms," in *Proc. 4th Indian Conf. Comput. Vis., Graph. Image Process.* India: Indian Unit for Pattern Recognition and Artificial Intelligence, 2004.

[37] G. W. Zack, W. E. Rogers, and S. A. Latt, "Automatic measurement of sister chromatid exchange frequency.," *J. Histochem. Cytochem.*, vol. 25, no. 7, pp. 741–753, Jul. 1977.

[38] A. W. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least squares fitting of ellipses," in *Proc. 13th Int. Conf. Pattern Recognit.*, Aug. 1996, pp. 253–257.

[39] X. Yu and E. Salari, "Pavement pothole detection and severity measurement using laser imaging," in *Proc. IEEE Int. Conf. Electro/Inf. Technol.*, Mankato, MN, USA, May 2011, pp. 1–5.

[40] S. Anand, S. Gupta, V. Darbari, and S. Kohli, "Crack-pot: Autonomous road crack and pothole detection," in *Proc. Digit. Image Comput., Techn. Appl. (DICTA)*, Dec. 2018, pp. 1–6.

[41] I. Schiopu, J. P. Saarinen, L. Kettunen, and I. Tabus, "Pothole detection and tracking in car video sequence," in *Proc. 39th Int. Conf. Telecommun. Signal Process. (TSP)*, Vienna, Austria, Jun. 2016, pp. 701–706.

[42] D. A. Casas Avellaneda and J. F. López-Parra, "Detection and localization of potholes in roadways using smartphones," *DYNA*, vol. 83, no. 195, pp. 156–162, 2016.

[43] W. G. Buttlar and M. S. Islam. (2014). *Integration of Smart-Phone-Based Pavement Roughness Data Collection Tool With Asset Management System—National Transportation Library*. Accessed: May 20, 2019. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/38287

[44] Roadroid. (2013). *Road Conditioning Monitoring Using Smart Phones. Quick Start Version 1.2.1*. Accessed: May 21, 2019. [Online]. Available: https://www.roadroid.com

[45] C. Chiculita and L. Frangu, "A low-cost car vibration acquisition system," in *Proc. IEEE 21st Int. Symp. Design Technol. Electron. Packag. (SIITME)*, Braşov, Romania, Oct. 2015, pp. 281–285.

[46] S. W. Katicha, J. El Khoury, and G. W. Flintsch, "Assessing the effectiveness of probe vehicle acceleration measurements in estimating road roughness," *Int. J. Pavement Eng.*, vol. 17, no. 8, pp. 698–708, 2015.

[47] A. Mednis, G. Strazdins, M. Liepins, A. Gordjusins, and L. Selavo, "RoadMic: Road surface monitoring using vehicular sensor networks with microphones," in *Networked Digital Technologies*, vol. 88. Berlin, Germany: Springer, 2010, pp. 417–429. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-14306-9_42

[48] W. Kongrattanaprasert, H. Nomura, T. Kamakura, and K. Ueda, "Detection of road surface states from tire noise using neural network analysis," *IEEJ Trans. Ind. Appl.*, vol. 130, no. 7, pp. 920–925, 2010.

[49] G. Erdogan, L. Alexander, and R. Rajamani, "Estimation of tire-road friction coefficient using a novel wireless piezoelectric tire sensor," *IEEE Sensors J.*, vol. 11, no. 2, pp. 267–279, Feb. 2011.

[50] I. Abdic, L. Fridman, D. E. Brown, W. Angell, B. Reimer, E. Marchi, and B. Schuller, "Detecting road surface wetness from audio: A deep learning approach," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Cancún, Mexico, Dec. 2016, pp. 3458–3463.

[51] J. Zhang, H. Qiu, S. S. Shamsabadi, R. Birken, and G. Schirner, "WiP abstract: System-level integration of mobile multi-modal multi-sensor systems," in *Proc. ACM/IEEE Int. Conf. Cyber-Phys. Syst. (ICCPS)*, Berlin, Germany, Apr. 2014, p. 227.

[52] Y. Zhang, J. G. McDaniel, and M. L. Wang, "Estimation of pavement macrotexture by principal component analysis of acoustic measurements," *J. Transp. Eng.*, vol. 140, no. 2, Feb. 2014, Art. no. 04013004.

[53] M. Wang, R. Birken, and S. Shahini Shamsabadi, "Implementation of a multi-modal mobile sensor system for surface and subsurface assessment of roadways," *Proc. SPIE*, vol. 9436, Mar. 2015, Art. no. 943607.

[54] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construct. Building Mater.*, vol. 157, pp. 322–330, Dec. 2017.

[55] D. Ai, G. Jiang, L. Siew Kei, and C. Li, "Automatic pixel-level pavement crack detection using information of multi-scale neighborhoods," *IEEE Access*, vol. 6, pp. 24452–24463, 2018.

[56] Y. Li, C. Papachristou, and D. Weyer, "Road pothole detection system based on stereo vision," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, Jul. 2018, pp. 292–297.

[57] U. Sinha. Labelling connected components—Example. AI Shack. Accessed: May 3, 2018. [Online]. Available: http://aishack.in/tutorials/labelling-connectedcomponents-example/

[58] L. Suong and K. Jangwoo, "Detection of potholes using a deep convolutional neural network," *J. Universal Comput. Sci.*, vol. 24, no. 9, pp. 1244–1257, 2018.

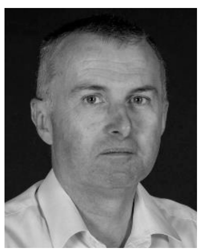
[59] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767

[60] C. Chellaswamy, H. Famitha, T. Anusuya, and S. B. Amirthavarshini, "IoT based humps and pothole detection on roads and information sharing," in *Proc. Int. Conf. Comput. Power, Energy, Inf. Commun. (ICCPEIC)*, Mar. 2018, pp. 084–090.

[61] S. W. Lee, S. Kim, J. Han, K. E. An, S.-K. Ryu, and D. Seo, "Experiment of image processing algorithm for efficient pothole detection," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–2.

JIHAD DIB received the B.Eng. degree (Hons.) in computer systems engineering from the University of Kent, Canterbury, U.K., in 2018. He is currently pursuing the Ph.D. degree in electronic engineering with the University of Kent. He is also a Research Ph.D. student with the School of Engineering and Digital Arts. His research interests are social assistive AI technologies, deep learning, pattern recognition, image processing, as well as robotics and machine learning. His awards and honours include a Ph.D. Scholarship, University of Kent V.S. Université Paris X Nanterre Robotics Competition (2nd Place), the IET Award for Best Overall Design and Performance Undergraduate 1st Year Robotics Competition (1st Place), the Judges Award for the Imaginative Software and Hardware Design in the Rapid Electronics Undergraduate 1st Year Robotics Competition, and other previous science and robotics competitions.

KONSTANTINOS SIRLANTZIS is currently a Lecturer in image processing and vision with the School of Engineering and Digital Arts, University of Kent, and also a Senior Lecturer in intelligent systems. He has a strong track record in image analysis and understanding, artificial intelligence, and neural networks for pattern recognition and biometrics-based security applications. He currently leads a Research Team working on social assistive and autonomous robotics and health-related assistive technologies. He has organized and chaired a range of international conferences and workshops, such as the RASC 2006 and Emerging Security Technologies 2010. He has authored over 100 peer-reviewed articles in journals and conferences.



GARETH HOWELLS is currently a Professor of secure electronic systems. He has been involved in research relating to security, biometrics, and pattern classification techniques for over 25 years and he has been awarded, either individually or jointly, several major research grants relating to the pattern classification and security fields, publishing over 180 papers in the technical literature. His recent work has been directed toward the development of secure device authentication systems with a focus on the Internet of Things (IoT), which has received significant funding from the a variety of sources. He has received, either individually or jointly, several major research grants relating to the pattern classification and security fields, and has authored over 180 papers in the technical literature. His recent work has been directed toward the development of secure device authentication systems which has received significant funding from several funding bodies and is currently in the process of being commercially exploited.