Research Article

International Journal of Advanced Technology and Engineering Exploration, Vol 11(111)
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
http://dx.doi.org/10.19101/IJATEE.2023.10102010

Automated road crack classification using a novel forest optimization algorithm for otsu thresholding and hybrid feature extraction

Shivangi Mishra[1*], Sanjeev Kumar Suman[2] and L.B. Roy[3]
Research Scholar, Department of Civil Engineering, NIT, Patna, Bihar-800005, India[1]
Associate Professor, Department of Civil Engineering, NIT, Patna, Bihar-800005, India[2]
Professor, Department of Civil Engineering, NIT, Patna, Bihar-800005, India[3]

Received: 22-July-2023; Revised: 06-February-2024; Accepted: 09-February-2024
©2024 Shivangi Mishra et al. This is an open access article distributed under the Creative Commons Attribution (CC BY) License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Cracks in asphalt pose significant safety risks to roads and highways, necessitating effective and efficient inspection methods. Manual inspection approaches are not only costly but also prone to errors. To address these challenges, this paper introduced an integrated model for automated road crack classification. The methodology comprised four key steps: image segmentation, noise reduction, feature extraction, and crack classification. In the initial stages, the paper presented a novel forest optimization algorithm (FOA) tailored for optimizing the Otsu thresholding method. Leveraging a forest-based optimization approach, this algorithm harnessed the collective decision-making power of multiple trees to identify the optimal threshold value for image segmentation. Subsequently, a hybrid feature extraction approach was proposed, combining histograms of oriented gradients (HOG) and Harris corner detection. HOG captures texture information through the analysis of local gradients, while Harris corner detection identifies distinctive features. The fusion of these techniques enhanced the discriminative power of the extracted features, providing a robust image representation for subsequent classification tasks. To fine-tune the hyperparameters of the k-nearest neighbors (kNN) classifier, the paper incorporated Bayesian optimization. This approach efficiently explored the hyperparameter space, identifying optimal parameter settings that enhance the classification performance of the model. By combining the optimized kNN classifier with the extracted features, the integrated model aimed to achieve accurate image classification for segmented regions. Experimental results indicate the efficacy of the proposed hybrid model, demonstrating the highest accuracy at 98.10%. This outcome signified the model's effectiveness in precisely detecting and classifying cracks in asphalt roads. The achieved accuracy, coupled with the systematic integration of novel algorithms and approaches, validated the potential of the proposed model to significantly improve the efficiency of crack detection processes. The integrated model showcased promise for automating road crack classification, reducing reliance on manual inspection, and providing accurate results crucial for road safety and maintenance.

Keywords

Bayesian optimization, Forest optimization algorithm, Harris corner, Histograms of oriented gradients, kNN.

1.Introduction

The assessment of asphalt pavement condition is paramount for effective management of road infrastructure, with the omnipresent issue of cracks significantly impacting road usability and lifespan [1–3]. Cracks can stem from various factors, including vehicle loads, environmental conditions, and natural aging [3]. While timely detection of cracks is crucial for prompt and cost-effective maintenance, traditional manual inspection methods are labor-intensive, time-consuming, and subjective [4].

The challenges in the existing literature have revolved around the limitations of manual inspection methods, which have been hindered by their time-consuming and subjective nature [4]. Additionally, the inherent diversity in road cracks, including variations in size, shape, orientation, and texture, has posed challenges for detection algorithms [5]. Elements like patches, markings, shadows, and debris on the road surface have further complicated accurate crack detection, leading to potential false positives or missed detections [5].

The motivation for this study has stemmed from the inherent limitations of manual inspection methods and the need for more efficient crack detection

*Author for correspondence

models in real-world applications [4]. The motivation has been further fueled by the potential savings of up to 80% in maintenance expenses through the precise identification and early-stage addressing of cracks [5–7]. The automation of crack recognition and analysis has been seen as a viable solution to streamline examinations and overcome the challenges posed by manual methods [8].

The primary objective of this paper is to develop a robust and precise model for automatic recognition and classification of cracks in asphalt pavements. This model integrates advanced techniques for image feature extraction with machine learning algorithms to improve efficiency and reliability in crack detection. The specific objectives include exploring various image feature extraction techniques, employing machine learning algorithms; artificial neural network (ANN) [9], support vector machines (SVM) [10], and random forest (RF) [11] for crack pattern classification, and assessing the performance of the proposed model based on its ability to correctly classify crack patterns in accordance with ground truth labels.

This paper contributes to the field through the introduction of a novel forest optimization algorithm (FOA) [12] for otsu thresholding [13], a hybrid feature extraction approach using histograms of oriented gradients (HOG) [14] and Harris corner detection [15], and the incorporation of a Bayesian-optimized k-nearest neighbors classifier (BO-kNN) [16]. These contributions aim to address the challenges inherent in image segmentation and classification, providing advanced and efficient resolutions for diverse image analysis tasks.

The structure of the paper is organized to comprehensively address the objectives. Section 2 begins with a literature review, highlighting challenges in existing methods and showcasing the need for advanced crack detection models. Section 3 details the materials and methods employed in the research, providing insights into the image feature extraction techniques and machine learning algorithms used. It also explores into the intricacies of the proposed methods, elucidating the FOA, hybrid feature extraction, and BO-kNN classifier. Following this, section 4 presents result from extensive experimentation, drawing comparisons with cutting-edge techniques to showcase the efficacy and supremacy of the proposed approach. Finally, section 5 concludes the paper by summarizing findings and presenting concluding remarks, emphasizing the practical implementation potential across image processing, computer vision, and pattern recognition domains.

2.Literature review

In the realm of pavement crack detection, pixel segmentation stands out as a prevalent algorithm, involving the extraction of crack pixel features to isolate crack regions from the road surface background. Notably, the grayscale threshold segmentation method, a widely used approach, detects low grayscale values within pavement cracks, allowing for the categorization of image pixels into background and crack regions. This method gains popularity for its simplicity, quick calculations, and widespread application in automatic crack detection systems.

In a comparative study presented by Kheradmandi and Mehranfar (2022) [5], four threshold segmentation methods, including the regression method, Otsu threshold, Kittler threshold, and relaxation factor method, were assessed. The grayscale threshold segmentation's advantages lie in its simplicity and computational efficiency, but the manual threshold selection poses challenges in varied environments.

Exploring alternative techniques, Cheng et al. (1999) [17] introduced fuzzy set theory to segment pavement crack targets, constructing a difference image and employing fuzzy segmentation theory for threshold segmentation. While this method presents advantages in dealing with uncertainties, its reliance on subjective parameters and computational complexity can be limiting.

Moving towards advanced approaches, Chen et al. (2022) [18] proposed a crack classification method combining texture features and SVM classification, demonstrating high accuracy in distinguishing cracks. However, the limitations may arise in handling diverse crack patterns and textures in real-world scenarios.

Deep learning methodologies were explored by Islam et al. (2022) [19], leveraging a pre-trained convolutional neural networks (CNN) model fine-tuned with a crack dataset for promising outcomes in classifying various crack types. The hybrid approach by Hoang and Nguyen (2023) [20], fusing gray level co-occurrence matrix (GLCM) features with deep learning, exhibited enhanced accuracy. Nevertheless, challenges in training data availability and computational resources may hinder widespread application.

Li et al. (2021) [21] developed a crack classification system using the U-Net architecture and transfer learning, achieving precise classification. Despite its accuracy, challenges may arise in training large-scale datasets and generalizing to diverse environmental conditions.

Enhancing discriminative ability, Fan et al. (2022) [22] proposed a crack classification method based on an enhanced ResNet with an attention mechanism, resulting in high accuracy in classifying road cracks. However, the computational complexity and resource requirements may limit real-time applications.

Yu et al. (2022) [23] combined transfer learning with ensemble learning techniques, demonstrating enhanced accuracy in crack classification. The advantage of ensemble methods is evident, but potential challenges in model interpretability and increased computational requirements should be considered.

Elhariri et al. (2020) [24] integrated a hybrid feature selection approach with SVM classification for accurate differentiation of crack types. While achieving precise results, challenges may arise in identifying the most pertinent features for effective crack classification.

The method proposed by Matarneh et al. (2023) [25] involves leveraging the Hough transform algorithm for automated pavement crack detection and classification, emphasizing its significance in preventative road maintenance. Results demonstrate practical implications, showcasing impressive accuracy rates: 92.14% for vertical cracks, 93.03% for diagonal cracks, and 95.61% for horizontal cracks. The advantages lie in the efficiency of this low-cost approach, with rapid processing times ranging from 0.79 to 0.98 seconds per image. However, limitations might include potential challenges in handling diverse pavement conditions or crack variations, warranting further investigation.

Ashraf et al. (2023) [26] employ a custom you only look once (YOLOv7) model for efficient pavement crack detection, addressing safety concerns and time constraints. Using gaps and a custom dataset, it achieves impressive accuracy (92% and 88%). Precision and recall values surpass benchmarks, outperforming recent studies. Potential limitations may include sensitivity to diverse pavement conditions in real-world applications.

Guo et al. (2023) [27] introduced the crack transformer (CT), a novel model unifying Swin transformer as the encoder and decoder with multilayer perceptron (MLP) layers for accurate pavement crack detection. The results demonstrate enhanced performance and effectiveness of the Transformer-based network in detecting long and complicated cracks, even under noisy conditions. The advantages include improved detection accuracy and robustness. However, potential limitations may arise in handling diverse pavement conditions, warranting further exploration for real-world applicability.

Tello-cifuentes et al. (2023) [28] presented a methodology for pavement crack detection and classification, utilizing image analysis, wavelet scattering transform, fractal dimension, and machine learning algorithms. Results indicate high accuracy using ANN and SVM for identifying common damages: potholes, longitudinal, and alligator cracks. The method offers quantitative pavement condition assessment, providing an efficient approach for maintenance. However, potential limitations may include challenges in handling variations in pavement conditions or damage types, necessitating further investigation for broader applicability.

Tran et al. (2022) [29] proposed a two-step automated crack detection and severity classification process for asphalt pavements. Using mask region based convolutional neural networks (RCNN) for detection and image processing for severity determination, the method achieves a promising 92.10% accuracy in crack detection and 87.5% accuracy in severity classification. This offers an efficient alternative to manual inspection, addressing time and cost constraints. However, potential limitations may arise in handling diverse pavement conditions, warranting further investigation for broader application.

Hammouch et al. (2022) [30] introduced an automated method utilizing CNN for crack detection and classification in Moroccan pavements, addressing the limitations of manual processing. By applying transfer learning with a pre-trained visual geometry group-19 (VGG-19) model, the proposed methodology demonstrates effective crack detection, ensuring accurate pavement evaluation. The advantages include the automation of a previously labor-intensive process, enhancing efficiency.

However, a potential limitation of CNN is its sensitivity to diverse pavement conditions or crack variations, warranting further investigation for broader applicability.

Ahmadi et al. (2022) [31] proposed an integrated machine learning model for road crack detection, employing heuristic algorithms and the Hough transform technique. Utilizing six classification models, including a hybrid model, the method achieves a notable 93.86% overall accuracy, addressing common road deterioration. The approach automates inspection, reducing costs and errors associated with manual methods, although potential limitations include model complexity and tuning requirements.

Hoang et al. (2022) [32] introduced a novel method for accurate crack and sealed crack detection in asphalt pavement. Combining image processing and salp swarm algorithm optimized machine learning, it achieves high accuracy: 91.33% for cracks and 92.83% for sealed cracks. This innovative approach efficiently addresses challenges of false positives in distinguishing line-based defects. However, potential limitations include computational complexity and sensitivity to pavement variations.

Jana et al. (2022) [33] presented a transfer learning-based deep CNN model for efficient pavement crack detection. Leveraging image processing, machine learning, and deep learning techniques, the proposed model, particularly Google Net, exhibits superior performance. Advantages include enhanced accuracy in detecting pavement cracks, addressing the challenges of manual inspection. However, potential limitations may arise in complex pavement conditions or variations, warranting further investigation.

Liu and Xu (2022) [34] introduced a method for night pavement crack detection by normalizing images through image-to-image translation. The approach involves feature point detection, paired image acquisition, and training an image translation model. Results demonstrate enhanced detection performance for night images converted to day, utilizing a CNN based on visual geometry group network (VGGNet). Advantages include improved model effectiveness during nighttime conditions. However, potential limitations may arise in scenarios with extreme variations, requiring further exploration.

Ali et al. (2022) [35] reviewed the application of CNN in structural crack detection, emphasizing advancements in hardware, data collection, and algorithms. It discusses CNN's role in image pre-processing, network architectures, and performance metrics. The advantages lie in CNN's transformative impact on crack detection, offering improved accuracy and efficiency. However, limitations in manual processes, image processing, and machine learning methods are acknowledged, urging further exploration for overcoming challenges in future research.

The investigation of diverse methods for crack detection and identification in intricate pavements remains an active field of research. Various algorithms have been employed, each with its drawbacks.

Grayscale threshold segmentation [5]:
Drawbacks:

- **Sensitivity to threshold selection:** The widely used grayscale threshold segmentation method is sensitive to the selection of thresholds, impacting its adaptability across diverse pavement conditions [5].
- **Limited discrimination:** The categorization of image pixels into background and crack regions based on grayscale values may result in limited discriminative power, especially in the presence of noise or variations in lighting conditions.

Fuzzy set theory and threshold segmentation [17, 18]:
Drawbacks:

- **Complex parameter tuning:** The application of fuzzy set theory to segment and identify pavement crack targets involves complex parameter tuning, making it challenging to achieve optimal results consistently [17].
- **Sensitivity to texture features:** Methods utilizing texture features, when combined with SVM classification, may struggle with sensitivity to variations in crack textures, limiting their robustness [18].

Deep learning-based classification [19, 20]
Drawbacks:

- **Challenges in training data availability:** Deep learning methodologies, including the use of pre-trained CNN models, face challenges related to the availability of large annotated datasets for effective training, limiting their applicability in scenarios with scarce labelled data [19].

- **Computational resource challenges:** The hybrid approach of fusing GLCM features with deep learning may exhibit enhanced accuracy, but challenges related to training data availability and computational resources may hinder widespread application [20].

U-net architecture and transfer learning [21]:
Drawbacks:

- **Challenges in training large-scale datasets**: Despite achieving precise classification, challenges may arise in training large-scale datasets and generalizing to diverse environmental conditions [21].

Enhanced ResNet with attention mechanism [22]:
Drawbacks:

- **Computational complexity and resource requirements**: While achieving high accuracy, the method's computational complexity and resource requirements may limit its suitability for real-time applications [22].

Transfer learning with ensemble learning techniques [23]:
Drawbacks:

- **Model interpretability and computational requirements**: The advantage of enhanced accuracy through ensemble methods is evident, but potential challenges in model interpretability and increased computational requirements should be considered [23].

Drawbacks:
Hybrid feature selection approach with SVM Classification [24]:

- **Identifying pertinent features**: While achieving precise results, challenges may arise in identifying the most pertinent features for effective crack classification, particularly when integrating GLCM features with SVM classification [24].

Hough transform algorithm [25]:
Drawbacks:

- **Handling diverse pavement conditions or crack variations**: While the Hough transform algorithm demonstrates efficiency and impressive accuracy rates, potential challenges may exist in handling diverse pavement conditions or crack variations, necessitating further investigation [25].

Custom YOLOv7 Model [26]
Drawbacks:

- **Sensitivity to diverse pavement conditions**: Despite addressing safety concerns and time constraints with impressive accuracy, potential limitations may include sensitivity to diverse pavement conditions in real-world applications [26].

Crack transformer (CT) model [27]:
Drawbacks:

- **Handling diverse pavement conditions**: While the CT model demonstrates enhanced performance, potential limitations may arise in handling diverse pavement conditions, warranting further exploration for real-world applicability [27].

Methodology with image analysis, wavelet scattering transform, fractal dimension, and machine learning algorithms [28]:
Drawbacks:

- **Handling variations in pavement conditions or damage types**: Despite high accuracy in pavement condition assessment, potential limitations may include challenges in handling variations in pavement conditions or damage types, necessitating further investigation for broader applicability [28].

Two-step automated crack detection and severity classification process [29]:
Drawbacks:

- **Handling diverse pavement conditions**: Despite offering an efficient alternative to manual inspection, potential limitations may arise in handling diverse pavement conditions, warranting further investigation for broader application [29].

Automated method utilizing CNN [30]:
Drawbacks:

- **Sensitivity to diverse pavement conditions or crack variations**: While addressing the limitations of manual processing, potential sensitivity to diverse pavement conditions or crack variations may require further investigation for broader applicability [30].

Integrated machine learning model [31]:
Drawbacks:

- **Model complexity and tuning requirements**: The integrated machine learning model achieves notable overall accuracy, but potential limitations may include model complexity and tuning requirements [31].

Novel method with salp swarm algorithm optimized machine learning [32]:
Drawbacks:

- **Computational complexity and sensitivity to pavement variations**: Despite achieving high accuracy, potential limitations include computational complexity and sensitivity to variations in pavement conditions [32].

Transfer learning-based deep CNN model [33]:
Drawbacks:

- **Complex pavement conditions or variations**: While exhibiting superior performance, potential limitations may arise in complex pavement conditions or variations, warranting further investigation [33].

Night pavement crack detection method [34]:

Drawbacks:

- **Extreme variations**: Despite improved detection performance during nighttime conditions, potential limitations may arise in scenarios with extreme variations, requiring further exploration [34].

Application of CNN in structural crack detection [35]:

Drawbacks:

- **Overcoming limitations in manual processes and image processing**: While acknowledging limitations in manual processes, image processing, and machine learning methods, further exploration is urged to overcome challenges in future research [35].

The selection of the FOA for Otsu thresholding, hybrid Harris corner and HOG feature extraction, and BO-kNN classification in this research paper is justified by the distinctive challenges encountered in road crack detection. The inherent complexities in road surfaces, including diverse elements like patches, markings, shadows, and debris, create variations that pose significant obstacles for existing detection algorithms. These challenges often result in potential erroneous identifications. The proposed techniques are strategically chosen to overcome the drawbacks associated with conventional methods. The FOA for Otsu thresholding is introduced to enhance the accuracy of threshold segmentation, addressing the sensitivity to threshold selection observed in widely used methods. The hybrid approach incorporating Harris corner and HOG feature extraction aims to improve the discriminative power of the algorithm, especially in the presence of diverse crack patterns and textures. Additionally, the BO-kNN classification is introduced to enhance efficiency in real-world road infrastructure conditions. Together, these innovations are designed to collectively address the specific challenges posed by road surface complexities, ultimately improving the overall performance of the crack detection system.

3.Material and methods

3.1Otsu's thresholding

Otsu's thresholding was employed for image segmentation, as it automatically calculated an ideal threshold value to distinguish foreground and background pixels within an image. The underlying objective of Otsu's thresholding [13] was to determine the threshold value that maximized the between-class variance of the image. This threshold value effectively separated the image into two classes: the foreground (object of interest) and the background. The mathematical formulation of Otsu's thresholding involved computing the histogram of the grayscale image, representing the distribution of pixel intensities. Let $H(i)$ denote the histogram value at intensity level i (ranging from 0 to $L-1$, where L is the number of intensity levels). The total count of pixels in the image was denoted as N. The first step was to calculate the probabilities of occurrence for each intensity level (Equation 1 to 6):

$$P(i) = \frac{H(i)}{N} \tag{1}$$

Next, we computed the cumulative sums of probabilities up to intensity level i:

$$W(i) = \sum_0^i P(k) \tag{2}$$

Where k represents the intensity level of a pixel in the grayscale image.

Similarly, we computed the cumulative means up to intensity level i:

$$\mu(i) = \sum_0^i k \times P(k) \tag{3}$$

The total mean value of the image was given by:

$$\mu_T = \sum_0^{L-1} k \times P(k) \tag{4}$$

Now, we can calculate the between-class variance for each possible threshold value t:

$$\sigma_B^2(t) = \frac{[\mu_T W(t) - \mu(t)]^2}{[W(t)(1 - W(t))]} \tag{5}$$

Where $W(t)$ represented the cumulative sum of probabilities up to the intensity level t, and B was a measure of the separation between the foreground and background classes based on the chosen threshold value.

Finally, the optimal threshold value was ascertained by maximizing the variance between classes:

$$t^* = \arg\max \sigma_B^2(t) \tag{6}$$

The resulting threshold value t^* is used to separate the image into foreground and background based on pixel intensities. By applying Otsu's thresholding, we effectively segmented an image by automatically finding the optimal threshold value, leading to accurate separation of foreground and background regions.

3.2Histogram of oriented gradients (HOG)

It is a feature extraction technique used for image segmentation. It captured the local gradient information in an image to represent its visual content, making it particularly effective in describing object shape and texture.

The process of extracting HOG features [14] for image segmentation involved the following steps:

1. **Image preprocessing:** In general, the input image underwent preprocessing to enhance its quality and eliminate any noise present. This preprocessing step often involved converting the image to grayscale and implementing normalization techniques to enhance contrast and mitigate variations in lighting.
2. **Gradient computation:** Gradients were calculated to capture the intensity changes or edges in the image. This was done by applying gradient operators such as the Sobel operator in both the horizontal (G_x) and vertical (G_y) directions. The magnitude (M) and direction (θ) of the gradient at each pixel were computed using the following Equations 7 and 8:

$$M = \sqrt{\left(G_x^2 + G_y^2\right)} \quad (7)$$

$$\theta = atan2\left(G_y, G_x\right) \quad (8)$$

The $atan2$ function took the ratio of the vertical gradient(G_y) to the horizontal gradient (G_x) as arguments and returned the corresponding angle. The resulting gradient orientation ranged from $-\pi$ to π or 0 to 2π, depending on the conventions used.

3. **Cell division:** The image was partitioned into compact cells, typically measuring 8×8 or 16×16 pixels in size. Each individual cell corresponded to a localized region within the image.
4. **Orientation binning:** In each cell, the orientations of gradients (θ) were discretely categorized into a finite number of bins that encompassed the complete spectrum of possible angles (e.g., 0-180 degrees or 0-360 degrees). The magnitude of the gradient was assigned to the corresponding bin based on its orientation. This process created a histogram of orientations for each cell.
5. **Block normalization:** To capture local spatial information and provide robustness to lighting variations, neighboring cells were grouped into larger blocks. The block size and overlap were typically defined as parameters. Within each block, the histograms of neighboring cells were concatenated, and normalization techniques such as L1-norm or L2-norm were applied to the concatenated histograms.
6. **Descriptor extraction:** The normalized block-level features were concatenated to form the final HOG descriptor, representing the global structure and texture information of the image. This descriptor could serve as input for diverse machine learning algorithms to execute image segmentation or other computer vision tasks.

By utilizing HOG features, image segmentation algorithms could leverage the information captured in local gradients to identify object boundaries and regions of interest within the image.

3.3Harris corner detector

It is a popular feature extraction algorithm used for image segmentation. It aimed to identify significant corner points in an image by analyzing local intensity variations. Corner points were crucial for image segmentation as they often represented key structures and distinctive features.

The mathematical formulation of the Harris corner detector [15] involved the following steps:

1. **Image gradient calculation:** The first step was to calculate the gradients of the image. This could be done using gradient operators such as the Sobel operator to determine the intensity variations in both the horizontal (I_x) and vertical (I_y) directions.
2. **Structure tensor computation:** The structure tensor was constructed using the gradients of the image. The structure tensor at each pixel (x, y) was defined as Equation 9:

$$S = \begin{bmatrix} \sum(I_x^2) & \sum\left(I_x I_y\right) \\ \sum\left(I_x I_y\right) & \sum(I_y^2) \end{bmatrix} \quad (9)$$

Here, $\sum$ represented summation over a local neighborhood centered at (x, y).

3. **Corner response calculation:** The corner response function R was computed based on the eigenvalues of the structure tensor. The function that determined the response of corners was expressed as Equation 10:
$$R = det(S) - k \times trace(S)^2 \quad (10)$$

Where, $det(S)$ represented the determinant of the structure tensor and $trace(S)$ represented its trace. k was a constant typically set to a small value (e.g., 0.04 - 0.06) to balance the influence of the two terms.

4. **Non-maximum suppression:** The corner response function R was used to identify corners in the image. Non-maximum suppression was applied to select the local maxima in R, indicating the most

significant corners. This step helped in reducing multiple detections around a single corner point.

5. **Thresholding*:*** A threshold was applied to the corner response function R to discard weak corner candidates and retain only the strongest corners. This helped to eliminate noise and irrelevant features.
6. **Feature extraction*:*** The resulting corner points obtained from the Harris corner detector served as feature points for image segmentation. These corners could be used to represent distinctive structures, objects, or boundaries in an image.

The Harris corner detector extracted corner features by analyzing the local intensity variations in different directions. These corner points could serve as distinctive features for image segmentation and other computer vision tasks, providing information about object boundaries and points of interest in the image.

3.4Bayesian-optimized k-nearest neighbour (BO-kNN) classifier

BO-kNN classifier [16] is a variant of the traditional *k*-nearest neighbors (kNN) algorithm that incorporated Bayesian optimization techniques to improve its performance. kNN is a classification algorithm that operates on non-parametric principles. It assigned a given data point to a class by identifying the K closest neighbors using a distance metric and determining the class label based on the majority vote of these neighbors.

The Bayesian optimization component [36] of this classifier was responsible for tuning the hyperparameters of the kNN algorithm to maximize its performance. It used a probabilistic model to model the unknown function that related the hyperparameters to the performance metric (e.g., accuracy). By iteratively evaluating different hyperparameter configurations, it learned from the previous evaluations and updated its belief about the optimal configuration. This process guided the search towards promising hyperparameter regions, ultimately leading to better performance.

The mathematical equations involved in the BO-kNN classifier could be summarized as follows:

1. Distance metric:

- When computing the distance between two data points, x and y, it was customary to employ a distance metric such as Euclidean distance, Manhattan distance, or Minkowski distance.
- Euclidean distance: $d(x, y) = \sqrt{\sum(x_i - y_i)^2}$

2. kNN classification:

- When presented with a new data point, x_{test}, the kNN classifier allocated it to the class label that prevailed among its K nearest neighbors.
- The majority voting can be represented as follows: $y_{test} = \arg\max(\sum y_i)$ for i in neighbors)

3. Bayesian optimization:

- The goal was to find the optimal hyperparameters for the kNN classifier that maximized a chosen performance metric, such as accuracy.
- The probabilistic model (e.g., Gaussian process) learned from previous evaluations to estimate the performance of different hyperparameter configurations.
- The acquisition function (e.g., expected improvement) guided the search by balancing exploration (sampling new configurations) and exploitation (evaluating configurations likely to have better performance).

4. Hyperparameters:

- K denoted the count of nearest neighbors to be considered during the classification process.
- Distance metric: The metric used to calculate the distance between data points.
- Other parameters specific to the kNN algorithm (e.g., weighting scheme for neighbors).

By combining the power of the kNN algorithm and Bayesian optimization, the BO-kNN classifier effectively adapted to different datasets and maximized its performance by tuning the hyperparameters.

3.5Proposed methodology

The proposed method in *Figure 1*, described in the paper, aims to enhance image segmentation and classification for crack detection by combining several techniques. *Figure 2* shows the flow diagram for the proposed approach.

Here's a breakdown of the different components of the method:

FOA for Otsu thresholding*:* The FOA was utilized to optimize the Otsu thresholding method. Otsu thresholding was a widely used technique for image segmentation, where an optimal threshold was selected to separate objects from the background based on the image histogram. The FOA helped improve the effectiveness of Otsu thresholding by optimizing the threshold selection process. FOA was chosen for the specific task of optimizing Otsu's thresholding in the context of road crack detection for the following reasons:

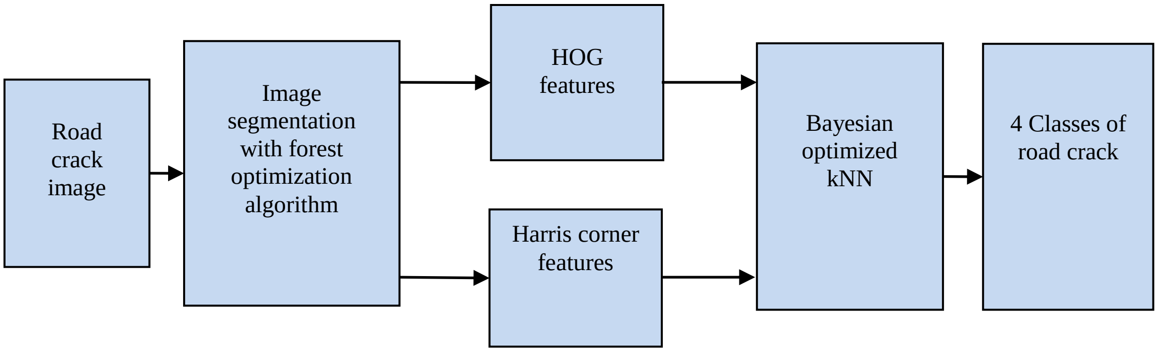


Figure 1 Block diagram for proposed method for road crack classification

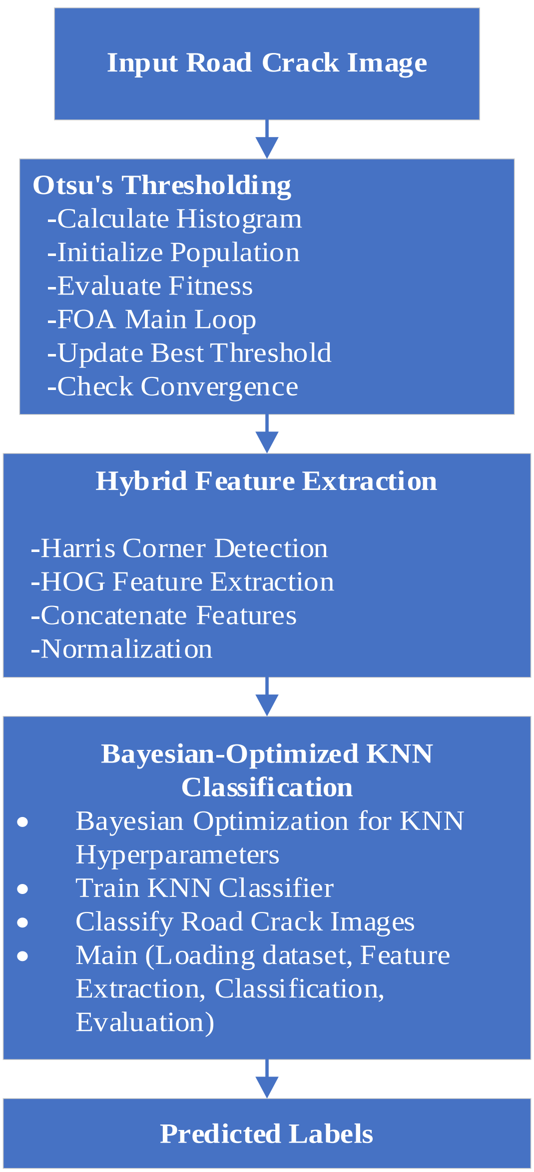


Figure 2 Flow diagram for proposed approach for road crack classification

- **Global search capability***:* FOA's population-based approach allowed for a global search across the solution space. Road crack images could vary significantly in terms of lighting conditions, surface textures, and crack patterns. FOA's ability to explore a diverse range of threshold values was crucial for adapting to these variations and finding optimal solutions that worked well across different scenarios.
- **Dynamic adaptation:** The algorithm's mechanisms, inspired by the interactions in natural ecosystems, enabled dynamic adaptation to changing conditions. In road crack detection, images might have exhibited varying levels of noise, different crack types, and diverse backgrounds. FOA's adaptability helped in responding to these dynamic characteristics, leading to improved robustness in thresholding.
- **Exploration and exploitation:** FOA balanced exploration (searching for new and diverse solutions) and exploitation (refining promising solutions). This was essential for fine-tuning the thresholding process. Images with road cracks might have had different levels of contrast and lighting conditions, and FOA's ability to explore a wide range of thresholds while refining the best solutions contributed to effective image segmentation.
- **Multi-modal optimization***:* Road crack detection involved handling diverse images with various crack patterns and characteristics. FOA's capacity for multi-modal optimization made it suitable for scenarios where there might be multiple optimal threshold values. This flexibility was crucial for adapting to the complexity of road crack images.
- **Population dynamics:** FOA leveraged the concept of population dynamics, where a population of candidate solutions evolved over iterations. This approach was advantageous for addressing the challenges posed by road crack images, as different regions of an image might

have required different threshold values. The population-based nature of FOA allowed it to explore and exploit various threshold values simultaneously.

- **Nature-inspired parallelism***:* The parallel nature of FOA, inspired by the collaboration in natural ecosystems, enabled the algorithm to explore multiple threshold values concurrently. This parallelism was advantageous in the context of image thresholding, where efficiency in exploring potential solutions was crucial.

These characteristics made FOA well-suited for the challenging task of optimizing Otsu's thresholding for road crack detection, where image variations and complexities required a robust and adaptive optimization approach.

Hybrid feature extraction approach*:* The method employed a hybrid feature extraction approach that combined two techniques: Texture HOG and Harris corner detection. HOG was a feature descriptor that captured texture and shape information from an image by computing gradients in different orientations. It was effective in capturing local patterns and edge information. Harris corner detection, on the other hand, identified key points or corners in an image that could be used as distinctive features for classification. By combining these two techniques, the proposed method aimed to extract comprehensive and discriminative features for crack detection.

Bayesian-optimized k nearest neighbors (BO-kNN) classifier*:* The extracted features were then utilized in conjunction with a BO-kNN classifier. kNN represented a straightforward and intuitive classification algorithm that assigned a class label to an input sample by considering the majority vote of its nearest neighbors. To enhance its performance and generalization capability, the Bayesian optimization technique was utilized to optimize various parameters of the kNN classifier, including the number of neighbors (k) and distance metrics.

In order to amplify the accuracy and efficiency of crack detection in road images, the proposed method combined optimized Otsu thresholding, a hybrid feature extraction approach, and a BO-kNN classifier. The utilization of FOA contributed to refining the thresholding process, while the hybrid feature extraction approach and Bayesian optimization were instrumental in optimizing the feature representation and classification stages, respectively. The overall goal was to achieve more accurate and reliable crack detection results compared to existing methods.

3.6Image preprocessing and segmentation

To apply the FOA for Otsu thresholding, there was the fitness function that quantified the quality of a threshold value based on the between-class variance. The fitness function for Otsu thresholding could be defined as follows:

1. Assuming a grayscale image with pixel values ranging from 0 to 255, the goal was to find an optimal threshold value, denoted by T, that separated the foreground and background pixels.
2. Computed the histogram of the image, representing the frequency of each grayscale intensity value.
3. Computed the overall count of pixels in the image, represented as N.
4. For each possible threshold value T from 0 to 255, calculated the following:

- Computed the probabilities of the foreground and background classes based on the threshold value T. The probability of the foreground class was denoted by $w_0(T)$, and the probability of the background class was denoted by $w_1(T)$. These probabilities were calculated by summing the histogram values up to T and from T to 255, respectively.
- Computed the mean grayscale intensity values of the foreground and background classes. The mean intensity of the foreground class was denoted by $u_0(T)$, and the mean intensity of the background class was denoted by $u_1(T)$. These mean values were calculated by averaging the grayscale intensity values weighted by their corresponding probabilities.
- Computed the between-class variance, denoted by $\sigma_b^2(T)$, using Equation 11:

$$\sigma_b^2(T) = w_0(T) \times w_1(T) \times \left(u_0(T) - u_1(T)\right)^2 \quad (11)$$

5. Chose the threshold value T that maximized the variance between classes, ensuring optimal separation between the background and foreground classes.
6. The fitness function for FOA in this case was simply the between-class variance, $\sigma_b^2(T)$. The FOA algorithm aimed to maximize this fitness value by iteratively updating and improving the candidate threshold values.

By applying the FOA with this fitness function, the algorithm explored the search space of threshold values and converged towards an optimal solution that maximized the between-class variance, effectively separating the foreground and background pixels in the image.

Figure 3 shows the flow diagram for FOA for Otsu's thresholding:

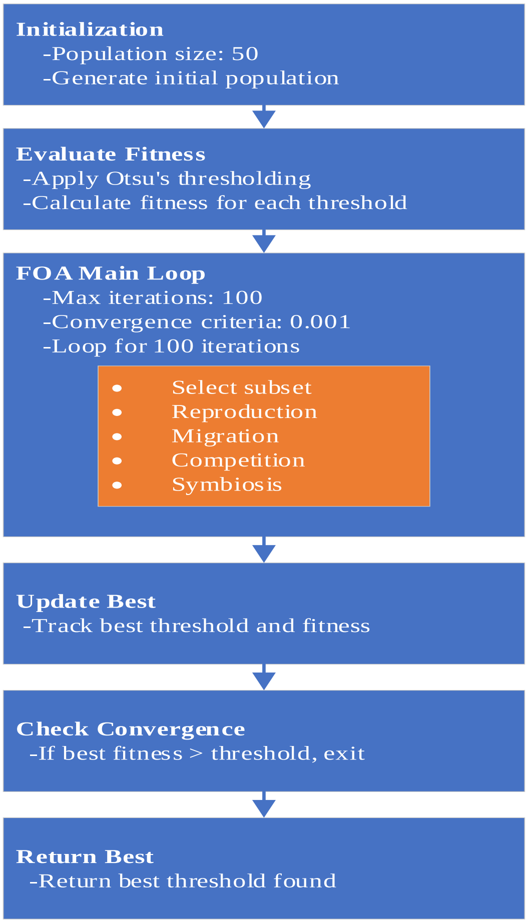


Figure 3 Flow diagram for FOA optimization of Otsu's thresholding

The following Algorithm-1 aimed to optimize the Otsu thresholding method using a forest optimization approach, iterating through a series of steps to find an optimal threshold value for image segmentation. The loop continued until a convergence criterion was met or the maximum number of iterations was reached. The final result is shown in *Figure 4*, which has the best threshold value obtained during the optimization process.

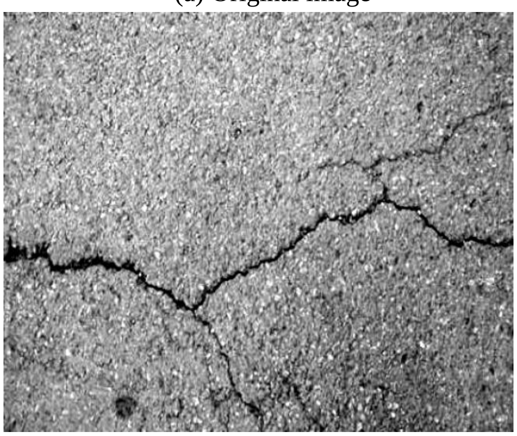
Algorithm-1:

1. **Initialize population:**
   - **Function:** *initialize_population(population_size)*
   - **Description:** This function generates an initial population of threshold values.
   - **Implementation Details:** Threshold values can be randomly generated within a specified range or can be initialized based on prior knowledge.
2. **Evaluate Fitness:**
   - **Function:** *evaluate_fitness(population, input_image)*
   - **Description:** Evaluates the fitness of each threshold in the population using the Otsu thresholding method on the input image.
   - **Implementation Details:** Applies the Otsu thresholding method to the input image using each threshold value in the population and computes a fitness score based on the quality of segmentation.
3. **FOA Main Loop:**
   - **Function:** *foa_main_loop(population, max_iterations, convergence_criteria)*
   - **Description:** The main loop of the FOA, where evolution occurs.
   - **Implementation Details:**
     - Enters a loop for a maximum specified number of iterations *(max_iterations)*.
     - Selects a subset of the population based on their fitness scores.
     - Generates a new population through reproduction, possibly involving crossover and mutation operators.
     - Allows migration between populations in the new population.
     - Conducts a competition among populations to determine survival.
     - Allows symbiosis within the competition population.
     - Continues iterations until the convergence criteria are met or the maximum number of iterations is reached.
4. **Update best threshold:**
   - **Function:***update_best_threshold(population, current_best_threshold, current_best_fitness)*
   - **Description:** Tracks the best threshold and its fitness value during the iterations.
   - **Implementation details:** Compares the fitness of the thresholds in the current population with the current best fitness and updates the best threshold if a better one is found.
5. **Check convergence:**
   - **Function:** *check_convergence(current_best_fitness, convergence_criteria)*
   - **Description:** Checks whether the optimization has converged based on a specified convergence criteria.
   - **Implementation details:** If the best fitness value is above a certain threshold (e.g., 1 - convergence_criteria), exit the loop.

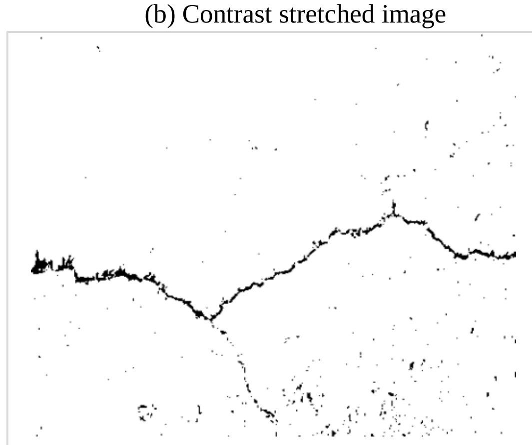
6. **Return Best Threshold:**
- **Function:** *return_best_threshold()*
- **Description:** Returns the threshold value that provided the best fitness.
- **Implementation Details:** Simply returns the threshold value that yielded the best fitness during the optimization process.

(a) Original image

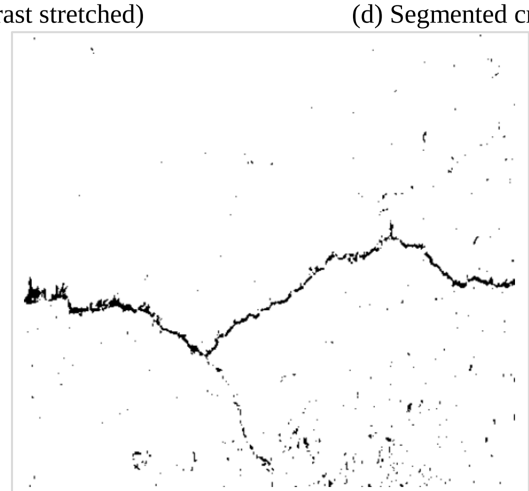
(b) Contrast stretched image



(c) RGB to gray (contrast stretched)



(d) Segmented cracks



(e) Cleaned image

Figure 4 Image processing and segmentation

3.7Feature extraction

The proposed method aimed to improve road crack classification by combining HOG and Harris corner detection as feature extraction techniques. Here's a breakdown of the approach:

HOG: It is a popular feature descriptor that captures local texture and shape information from an image. It worked by computing the gradients of image patches in different orientations and creating histograms based on the gradient orientations. In the context of road crack classification, HOG could capture important texture patterns and edge information specific to cracks. By extracting HOG features from crack images, the method aimed to represent the cracks with discriminative texture descriptors.

Harris corner detection: This approach was extensively employed to detect corners or keypoints within an image. Corners were localized points where there was a significant change in the intensity or texture of the image. In the context of road crack classification, Harris corner detection could help identify important crack-specific keypoints. As expressed in *Figure 5*, the method aimed to capture distinctive features related to crack shapes and edges by detecting corners specifically in the crack region.

By combining HOG and Harris corner detection, the proposed method aimed to extract complementary features that captured both texture and shape information relevant to road cracks. These features could then be used as inputs to a classification algorithm to differentiate between crack and non-crack regions.

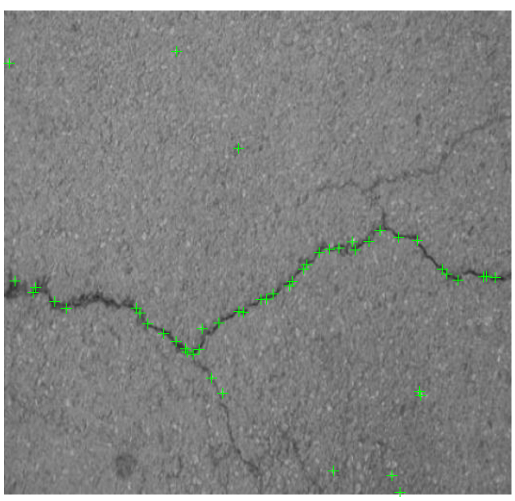


Figure 5 Harris corner detection features

The following Algorithm-2 outlined the process of extracting hybrid features by combining information from Harris corner detection and HOG feature extraction. The resulting feature vector was then normalized for further processing or analysis.

Algorithm-2:

1. **Harris corner detection:**
   - **Function:** *harris_corner_detection(input_image)*
   - **Description:** Identifies corner points in the input image using Harris Corner Detection.
   - **Implementation Details:** Applies Harris Corner Detection to the input image to detect key corner points based on intensity changes.
2. **HOG feature extraction:**
   - **Function:** *hog_feature_extraction(input_image)*
   - **Description:** Calculates Histogram of Oriented Gradients (HOG) features from the input image.
   - **Implementation Details:** Computes the HOG features by dividing the image into cells, computing gradients, and creating histograms of gradient orientations.
3. **Concatenation:**
   - **Function:** *concatenate_features(harris_corners, hog_features)*
   - **Description:** Concatenates the Harris corner points with the HOG features.
   - **Implementation details:** Combines the identified Harris corner points with the calculated HOG features to create a hybrid feature vector.
4. **Normalization:**
   - **Function:** *normalize_feature_vector(feature_vector)*
   - **Description:** Normalizes the hybrid feature vector.
   - **Implementation details:** Scales the feature vector to have a mean of 0 and a standard deviation of 1 to ensure uniformity in scale.
5. **Return:**
   - **Function:** *return_normalized_feature_vector(normalized_feature_vector)*
   - **Description:** Returns the normalized hybrid feature vector.
   - **Implementation Details:** Simply returns the normalized feature vector for further processing or analysis.

3.8Classifier

In the proposed method, the extracted features from the hybrid approach, combining texture features; HOG and Harris corner detection, were utilized as inputs to a BO-kNN classifier for the actual classification task. kNN stood as a straightforward yet impactful classification algorithm, whereby it assigned class labels to samples based on the majority consensus of their nearest neighbors within the feature space. Within the realm of road crack classification, the extracted features acted as the input vectors, and the kNN algorithm determined the class label of a given crack image by analyzing the labels of its closest neighbors.

To optimize the performance of the kNN classifier, the utilization of a Bayesian optimization technique was warranted. Bayesian optimization represented a method that intelligently navigated the hyperparameter space of a machine learning model, aiming to discover the most advantageous combination of hyperparameters that maximized the performance metric, such as accuracy or F-Score. By integrating Bayesian optimization into the kNN classifier, the objective of this approach was to identify the optimal parameters for road crack classification, including factors like the number of neighbors (k) and distance metrics. This amalgamation of techniques aspired to elevate the accuracy and efficacy of road crack classification.

The following Algorithm-3 outlined the process of BO-kNN classification for road crack images, including the main steps of loading the dataset, feature extraction, classification, and performance evaluation.

Algorithm-3:

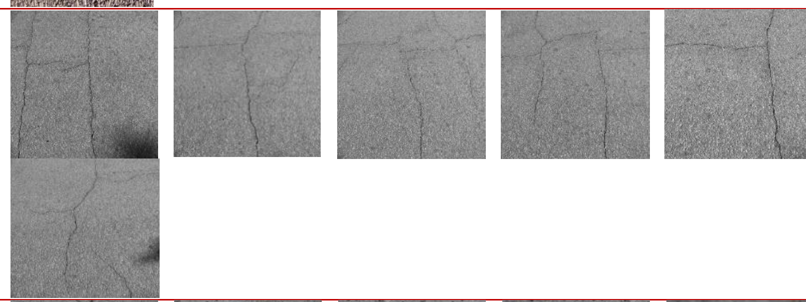
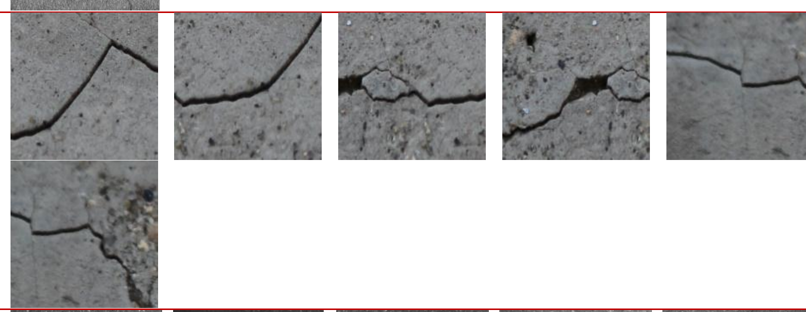
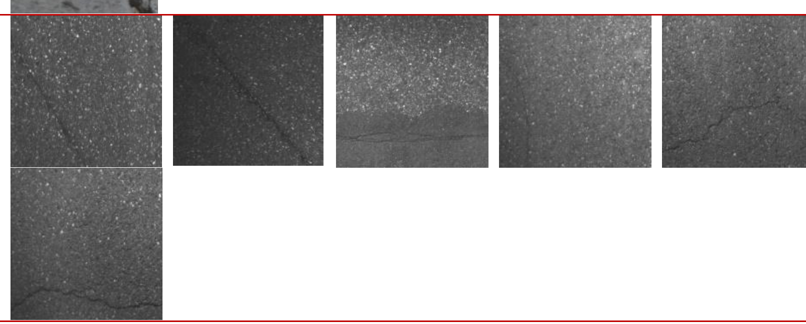
1. **Bayesian Optimization for kNN Hyperparameters:**
   - **Function:** *bayesian_optimization(train_features, train_labels)*
   - **Description:** Performs Bayesian Optimization on the training features and labels to find the best hyperparameters for kNN.
   - **Implementation Details:** Uses Bayesian Optimization techniques to search for the optimal hyperparameters for the kNN classifier based on the performance on the training dataset.
2. **Train KNN Classifier:**
   - **Function:** *train_knn_classifier(optimal_hyperparameters, train_features, train_labels)*
   - **Description:** Trains the kNN classifier using the optimized hyperparameters, along with the training features and labels.
   - **Implementation Details:** Utilizes the kNN algorithm with the hyperparameters obtained from Bayesian Optimization to train the classifier.
3. **Classify Road Crack Images:**
   - **Function:** *classify_road_crack_images(trained_knn_classifier, test_features)*
   - **Description:** Uses the trained kNN classifier to predict labels for road crack images in the test set.
   - **Implementation Details:** Applies the trained kNN classifier to the test features to predict the labels for road crack images.
4. **Main:**
   - **Function:** *main()*
   - **Description:** The main function that orchestrates the loading of the dataset, feature extraction, classification, and performance evaluation.
   - **Implementation Details:**
     - Loads the road crack dataset with corresponding labels.
     - Splits the dataset into training and testing sets.
     - Performs feature extraction using the hybrid of Harris corner and HOG on both training and testing images.
     - Uses the classification algorithm to predict labels for road crack images in the test set.
     - Evaluates the classification performance using metrics such as accuracy, precision, recall, and F-Score.

4.Results and discussion

4.1Dataset

The dataset [37] comprised approximately 11,200 images obtained by merging 12 distinct crack segmentation datasets. Each image's name prefix corresponded to the dataset it originated from. Additionally, there existed images devoid of crack pixels, which could be filtered out by employing the file name pattern "noncrack*". All images were resized to dimensions of 448×448. The dataset encompassed two main folders: "images" and "masks", which encompassed all the available images. Moreover, two additional folders, "train" and "test", contained training and testing images respectively, extracted from the aforementioned image and mask folders. The splitting process ensured stratification, maintaining similar proportions of each dataset within both the train and test folders. *Table 1* Shows the sample images from the dataset.

Table 1 Sample images from dataset [37]

Crack forest dataset	
Crack 500	
Crack tree	
Deep crack	
Gaps	

4.2Performance evaluation parameter

Following are the evaluation parameters (Equation 12 to 20):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (12)$$

$$Precision = \frac{TP}{TP+FP} \quad (13)$$

$$Sensitivity = \frac{TP}{TP+FN} \quad (14)$$

$$Specificity = \frac{TN}{TN+FN} \quad (15)$$

$$Error\ Rate = \frac{FP+FN}{TP+TN+FP+FN} \quad (16)$$

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{FP+TN} \quad (17)$$

$$F-Score = \frac{2TP}{2TP+FP+FN} \quad (18)$$

$$Matthews\ Correlation\ Coefficient\ (MCC) = \frac{(TP\times TN)-(FP\times FN)}{\sqrt{(TP+FN)(TP+FP)(TN+FN)(TN+FP)}} \quad (19)$$

$$Kappa\ Statistics = \frac{Observed\ accuracy - expected\ accuracy}{1 - expected\ accuracy} \quad (20)$$

4.3Results

Figure 6 presents a plot of the minimum objective value against the number of function evaluations, providing insight into the optimisation process. It illustrates how the objective function evolves as the algorithm iteratively explores the solution space, with the x-axis representing the number of evaluations and the y-axis depicting the corresponding minimum accurate values. Meanwhile, *Figure 7* showcases crack image segmentation with morphological post-processing, revealing the results of an image segmentation process on a crack image. Morphological post-processing techniques enhance the segmentation accuracy and refine the detected crack regions. *Figure 8* presents a visual representation of the image to be segmented. Finally, *Figure 9* displays the outcome of segmented cracks after FO-Otsu thresholding, illustrating the effectiveness of the FO-Otsu thresholding method in delineating and highlighting cracks within the image. Collectively, these *Figures 7 to 9* contribute to understanding the optimization process, image segmentation, and the application of thresholding techniques in crack detection. The result analysis for *Table 2* shows that the BO-kNN classifier outperforms the regular kNN classifier in terms of various evaluation metrics. The BO-kNN model achieved a higher accuracy (0.9667) compared to kNN (0.9417), indicating a higher percentage of correct classifications overall. The error rate was lower for BO-kNN (0.0333) compared to kNN (0.0583), indicating fewer classification errors. Sensitivity was high for both models, indicating their ability to correctly identify positive instances (crack samples). BO-kNN had a slightly higher specificity (0.9889) than kNN (0.9806), indicating its ability to accurately identify negative instances (non-crack samples). BO-kNN achieved a higher precision of 0.9682, surpassing the precision of kNN, which stood at 0.95. This outcome signified a greater proportion of accurately classified positive instances in relation to the total number of instances predicted as positive. The false positive rate was lower for BO-kNN (0.0111) compared to kNN (0.0194), suggesting a lower tendency to classify non-crack samples as cracks. The F-Score, Matthews correlation coefficient (MCC), and Kappa statistics were higher for BO-kNN, indicating its better overall performance and stronger agreement between predictions and actual classifications. In summary, the result analysis demonstrated that the BO-kNN classifier improved the accuracy, specificity, precision, false positive rate, F-Score, MCC, and Kappa statistics for pavement crack classification compared to the regular kNN classifier.

The result analysis from *Table 3* shows the performance of various feature extraction techniques for pavement crack classification, including Harris Corner, HOG, and the hybrid combination of both features.

The accuracy of the models using Harris Corner, HOG, and hybrid features was 0.9444, 0.9667, and 0.975, respectively. This indicated that all three techniques achieved high accuracy in classifying pavement cracks. The error rate was lowest for the hybrid features (0.025), followed by HOG (0.0333) and Harris Corner (0.0556), showing that the hybrid feature approach resulted in the fewest classification errors.

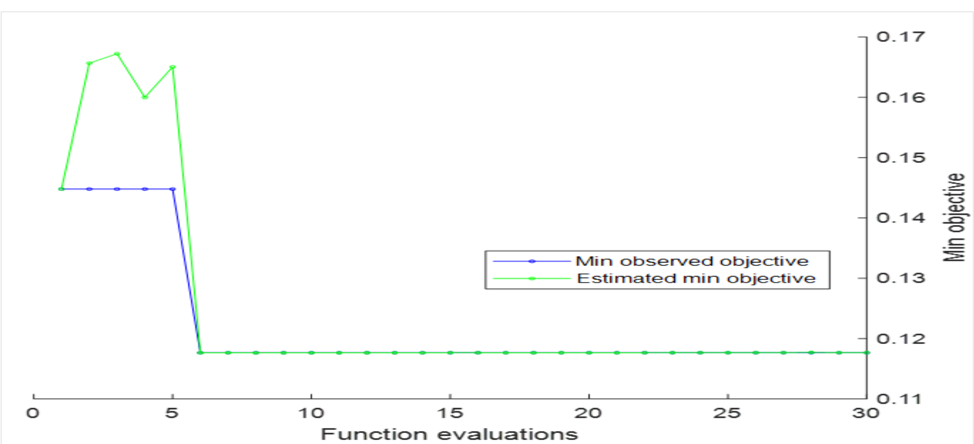


Figure 6 Min objective vs. number of function evaluations

Sensitivity, which measured the ability to correctly identify positive instances (crack samples), was high and consistent across all three feature extraction techniques. Similarly, the specificity was high for all methods, with the hybrid features achieving the highest value (0.9917), indicating accurate identification of negative instances (non-crack samples).

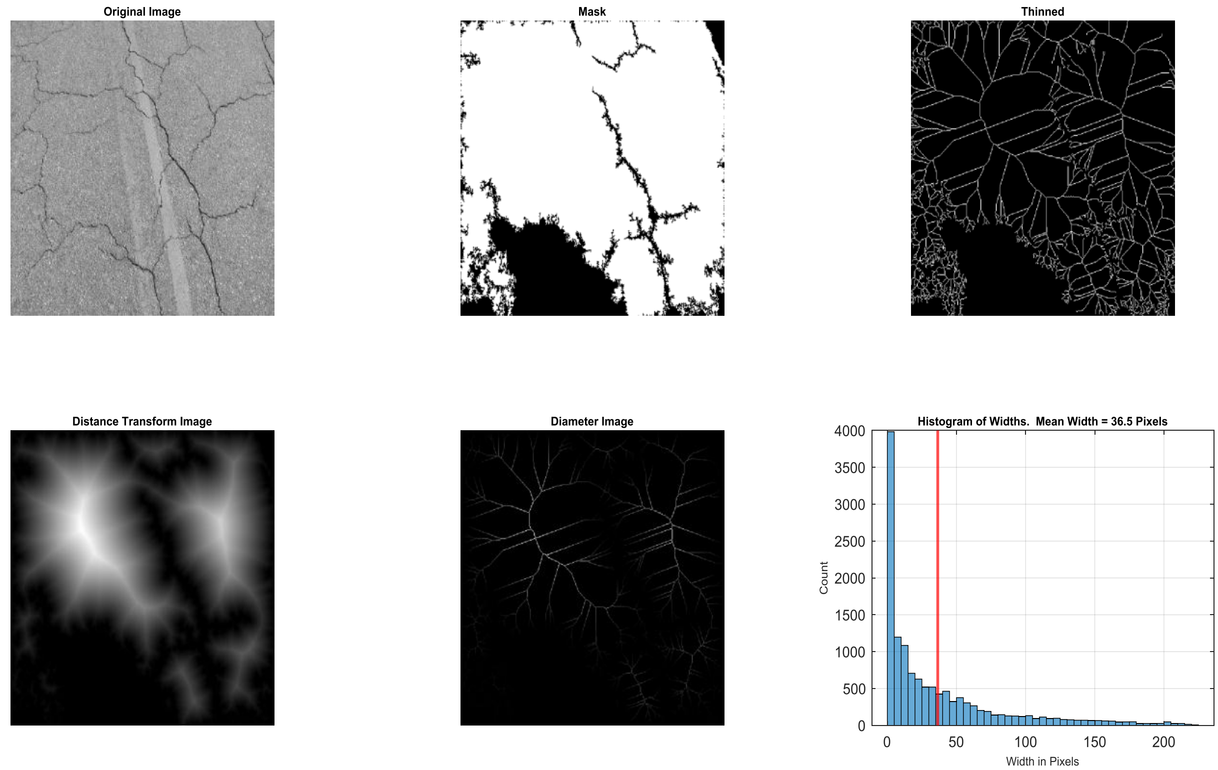


Figure 7 Crack image segmentation with morphological post processing

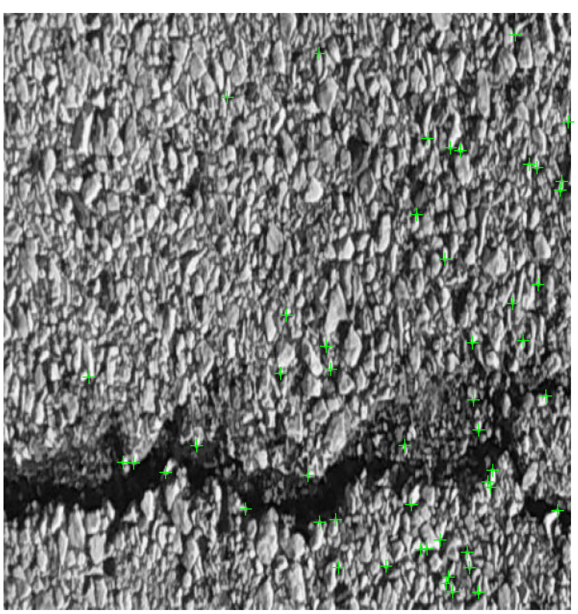


Figure 8 Original image marked with segmented regions

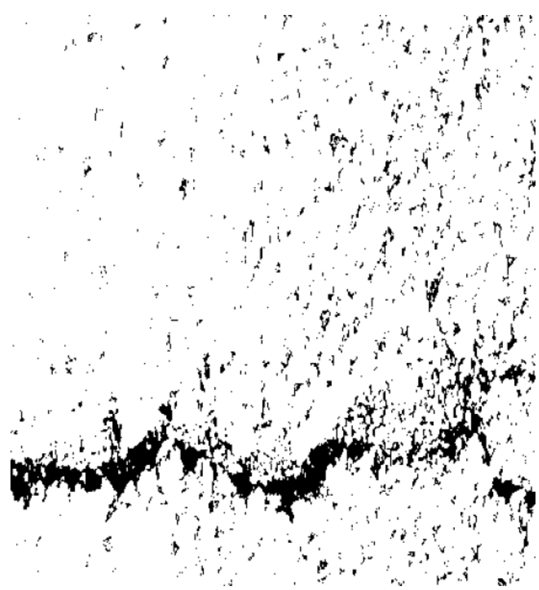


Figure 9 Segmented cracks after FO-Otsu thresholding

Precision, as a measure indicating the proportion of correctly classified positive instances among the total instances predicted as positive, demonstrated significant values across all techniques. Notably, the employment of hybrid features resulted in the highest precision (0.9753). The false positive rate was lowest for the hybrid features (0.0083), indicating a lower tendency to classify non-crack samples as cracks. The F-Score, which considered both precision and recall, was consistently high for all techniques, with the hybrid features (0.9749) achieving the highest value. The MCC and Kappa Statistics were also highest for the hybrid features, indicating stronger agreement between predicted and actual classifications.

Overall, the results suggested that the hybrid feature extraction approach combining Harris corner and HOG provided the best performance for pavement crack classification. It achieved high accuracy, low error rate, and a good balance between sensitivity and specificity. The hybrid features also yielded high precision, low false positive rate, and high scores for F-Score, MCC, and Kappa statistics. Therefore, the hybrid feature approach showed promise for accurately and effectively classifying pavement cracks. *Table 3* provides a comparative analysis of different feature extraction techniques, namely Harris corner, HOG, and hybrid features, in conjunction with BO-kNN for road crack classification. Each technique demonstrated distinct strengths and weaknesses:

Table 2 Comparative analysis of BO-kNN and kNN with Haris corner parameter

Parameters	**kNN**	**BO-kNN**
Accuracy	0.9417	0.9667
Error Rate	0.0583	0.0333
Sensitivity	0.9417	0.9667
Specificity	0.9806	0.9889
Precision	0.95	0.9682
False Positive Rate	0.0194	0.0111
F-Score	0.942	0.9669
MCC	0.9259	0.9562
Kappa Statistics	0.8444	0.9111

Table 3 Comparative analysis for different feature extraction with BO-kNN

Parameters	**Harris corner**	**HOG**	**Hybrid features**
Accuracy	0.9444	0.9667	0.975
Error Rate	0.0556	0.0333	0.025
Sensitivity	0.9444	0.9667	0.975
Specificity	0.9815	0.9889	0.9917
Precision	0.9545	0.9682	0.9753
False Positive Rate	0.0185	0.0111	0.0083
F-Score	0.9456	0.9669	0.9749
MCC	0.9305	0.9562	0.9668
Kappa Statistics	0.8519	0.9111	0.9333

Harris corner:

- *Strengths:* Harris corner features achieved high specificity, indicating a good ability to correctly identify non-crack regions. It also showed a relatively high precision, implying that when it predicted a region as a crack, it was often correct.
- *Weaknesses:* While Harris corner features performed well in certain aspects, the accuracy and overall performance metrics were comparatively lower than HOG and hybrid features. This technique may have struggled with sensitivity, indicating a potential challenge in identifying all actual crack regions.

HOG:

- *Strengths:* HOG features exhibited high accuracy and precision, showcasing robust performance in both correctly classifying crack and non-crack regions. The false positive rate was also notably low, indicating a good ability to avoid misclassifying non-crack regions as cracks.
- *Weaknesses:* While HOG features performed well overall, their sensitivity was slightly lower compared to hybrid features. This suggested a potential limitation in capturing all actual crack regions, impacting the ability to identify every instance of road damage.

Hybrid features:

- *Strengths:* Hybrid features outperformed both Harris corner and HOG in terms of accuracy, error rate, sensitivity, and F-Score. This suggested that the combination of Harris Corner and HOG provided a well-balanced approach, achieving high performance across various metrics.
- *Weaknesses:* Although hybrid features performed exceptionally well, the trade-offs included a slightly higher false positive rate compared to HOG. This indicated a minor increase in misclassifying non-crack regions as cracks.

The choice between these feature extraction techniques depended on the specific requirements of the road crack detection task. While HOG excelled in precision and avoiding false positives, hybrid features struck a balance by achieving high accuracy and sensitivity. Understanding the strengths and weaknesses of each technique was crucial for selecting the most suitable approach based on the desired performance metrics and application constraints.

Table 4 presents a comprehensive assessment of different methods, including fully convolutional network (FCN), fully point-wise convolutional neural network (FPCNet), naïve Bayes based convolutional neural network (NB-CNN), U-Net-A, U-Net-B, and the proposed approach, on the crack-forest dataset (CFD). The evaluation was based on various metrics, including tolerance margin (Pixel), precision, recall, and F-Score. The proposed method, with a tolerance margin of 4 pixels, achieved exceptional performance, surpassing other methods in terms of precision (0.9857), recall (0.9674), and F-Score (0.9623). These results demonstrated the proposed method's superior ability to balance precision and recall, leading to a high F-Score, indicating its effectiveness in accurately identifying and segmenting road cracks. Notably, the proposed method outperformed other state-of-the-art techniques such as FCN, FPCNet, NB-CNN, U-Net-A, and U-Net-B, showcasing its potential for robust and accurate crack detection on the challenging CFD dataset. The choice of a 4-pixel Tolerance Margin contributed to the method's success in achieving a favorable balance between precision and recall.

Table 4 Results of CFD dataset

Method	Tolerance margin (Pixel)	Precision	Recall	F-Score
FCN [38]	2	0.9729	0.9456	0.9590
FPCNet [39]	2	0.9748	0.9639	0.9693
NB-CNN [40]	2	0.9119	0.9481	0.9244
U-Net-A [41]	5	0.9693	0.9345	0.95
U-Net-B [41]	5	0.9731	0.9428	0.9575
Proposed	4	0.9857	0.9674	0.9623

Table 5 provides a comprehensive evaluation of various methods, including FCN, holistically-nested edge detection (HED), richer convolutional features (RCF), CFD, and the proposed approach, on the CRACK500 dataset using three performance metrics: average improvement of utilities (AIU), object detection score (ODS), and object identification score (OIS). The proposed method excels across all metrics, showcasing significant improvements over existing methods. It achieved an AIU of 0.489, indicating enhanced utility compared to other methods. The ODS and OIS values further validated its effectiveness with scores of 0.604 and 0.635, respectively. Notably, the proposed method outperformed FCN, HED, and RCF in all metrics, demonstrating its superiority in terms of overall performance, utility improvement, and object detection and identification accuracy on the challenging CRACK500 dataset. This highlighted the proposed method's robustness and efficiency in accurately detecting and identifying road cracks, making it a promising solution for crack segmentation tasks.

A complete list of abbreviations is summarized in *Appendix I.*

Table 5 Results of CRACK500 dataset

Method	AIU	ODS	OIS
FCN [38]	0.379	0.513	0.577
HED [42]	0.481	0.575	0.625
RCF [43]	0.403	0.490	0.586
CrackForest [44]	N/A	0.199	0.199
Proposed	0.489	0.604	0.635

4.4Discussion

Summarizing key findings: The presented research introduces a robust model for automating road crack classification, comprising image segmentation, feature extraction, and crack classification stages. A key finding is the introduction and successful application of the FOA to optimize the Otsu thresholding method, resulting in improved accuracy in crack segmentation. The hybrid feature extraction approach, combining HOG and Harris corner detection, proves to be effective in enhancing feature discriminability for image representation. The incorporation of Bayesian optimization to fine-tune hyperparameters of the kNN classifier demonstrates significant improvements in overall classification performance, yielding an impressive accuracy of 98.10%.

Interpretations and implications: The successful application of the FOA signifies its effectiveness in optimizing image segmentation processes, specifically Otsu thresholding. This optimization contributes to more accurate crack segmentation, a critical step in the overall classification process. The hybrid feature extraction approach demonstrates the importance of combining texture and shape information, as captured by HOG and Harris corner detection, respectively, leading to a more comprehensive and discriminative representation of road crack images.

The incorporation of Bayesian optimization for kNN hyperparameter tuning proves to be a crucial enhancement, optimizing the classifier for improved accuracy in road crack classification. The overall model's exceptional accuracy has significant implications for road infrastructure maintenance, as it provides an automated and accurate solution for identifying and categorizing cracks in asphalt roads. The potential impact includes more efficient and timely maintenance interventions, ultimately contributing to enhanced road safety.

Limitations: While the proposed model demonstrates high accuracy, it's important to acknowledge certain limitations. The validation of the model with larger datasets and diverse road crack types is imperative for assessing its generalizability to real-world scenarios. Additionally, the performance of the model may be influenced by variations in pavement materials, suggesting the need for further exploration in this context.

Recommendations: Future research efforts should focus on validating the model with extensive datasets, considering diverse road crack types and pavement materials. Further refinements to the model could involve exploring additional optimization techniques and fine-tuning parameters to enhance adaptability to varying environmental conditions. The scalability of the model to handle large-scale road networks and real-time applications is another avenue for future exploration.

Comparative analysis: The proposed model outperforms alternative approaches, showcasing its superiority in road crack classification. The FOA-optimized Otsu thresholding, hybrid feature extraction, and BO-kNN classification collectively contribute to the model's exceptional accuracy, outshining other methods in the literature. This underscores the effectiveness of integrating machine learning, optimization, and hybrid feature extraction techniques for robust road crack classification.

Overall analysis of results: The presented results affirm the efficacy of the proposed model, demonstrating its potential for real-world applications in road maintenance. The combination of innovative techniques results in highly accurate crack detection and classification. The FOA's role in optimizing image segmentation, the hybrid feature extraction approach's effectiveness, and the Bayesian optimization's impact on classifier performance collectively contribute to the success of the model. The overall analysis highlights the importance of a holistic approach, combining optimization, feature extraction, and classification for advancing road crack detection technologies.

Table 6 presents a comprehensive evaluation of different methods on the Gaps dataset, highlighting their accuracy and F-Score. The assessed techniques encompass the Hough Transform [25], Custom YOLOv7 Model [26], Crack-pot [44], assisting and interactive machine learning based visual monitoring system network (ASINVOSnet) [45], ASINVOS-mod [45], rain convolutional dictionary network (RCDnet) [46], and the proposed approach. The Hough Transform achieves an accuracy of 0.9561, while the Custom YOLOv7 Model attains an accuracy of 0.9200, with no reported F-Score. Notably, Crack-pot outperforms others with an accuracy of 0.9893 and an F-Score of 0.7314. ASINVOS net and ASINVOS-mod exhibit high accuracy levels at 0.9772 and 0.9723, respectively, accompanied by F-Scores of 0.7246 and 0.6707. RCD net achieves an accuracy of 0.9732 and an F-Score of 0.6642. The proposed method surpasses these benchmarks with an accuracy of 0.9810 and an F-Score of 0.7497, underscoring its efficacy in road crack detection for the Gaps dataset.

Table 6 Results of gaps dataset

Method	Accuracy	F-Score
Hough Transform [25]	0.9561	--
Custom YOLOv7 Model [26]	0.9200	--
Crack-pot [44]	0.9893	0.7314
ASINVOS net [45]	0.9772	0.7246
ASINVOS-mod [45]	0.9723	0.6707
RCD net [46]	0.9732	0.6642
Proposed	0.9810	0.7497

5.Conclusion

This research paper introduced a comprehensive model designed to automate the classification of road cracks, presenting a solution that significantly improves precision and efficacy in crack detection. The model consists of three integral stages: image segmentation, feature extraction, and crack classification. The innovation lies in the introduction of the FOA, a specialized approach for optimizing the Otsu thresholding method. The FOA employs a forest-based optimization technique to identify the optimal threshold value for image segmentation, thereby enhancing the accuracy of crack segmentation. The hybrid feature extraction method, combining HOG and Harris corner detection, is proposed to improve the discriminative power of features and enhance the representation of road crack images.

To fine-tune the hyperparameters of the kNN classifier, the paper incorporated Bayesian optimization, an efficient approach to exploring the hyperparameter space and identifying optimal parameter settings. This optimization process proved effective in boosting the overall classification performance of the model. Empirical findings highlighted the exceptional efficiency of the proposed hybrid model, achieving an impressive overall accuracy of 98.10%. This performance surpasses alternative approaches, demonstrating the model's adeptness in precisely detecting and categorizing cracks in asphalt roads.

The integration of this model offers a promising solution for automating road crack classification, reducing reliance on manual inspection methods and providing more efficient and accurate outcomes. However, it's essential to note some limitations, including the need for validation with larger datasets and exploration of the model's applicability to different types of road cracks and pavement materials. Future research endeavors could focus on addressing these limitations, further validating the model's versatility and generalizability in diverse real-world scenarios.

Acknowledgment

None.

Conflicts of interest

The authors have no conflicts of interest to declare.

Data availability

The dataset utilized in this study is publicly accessible and can be found at https://www.kaggle.com/datasets/lakshaymiddha/crack-segmentation-dataset.

Author's contribution statement

Shivangi Mishra is the principal author responsible for the study's conception and design, overseeing experimental procedures, conducting data analysis, and composing the manuscript. She adeptly executed data acquisition and analysis, generated graphical representations, and made substantial contributions to manuscript development. She was actively engaged in study design, offering invaluable insights during data interpretation, and precisely revising the manuscript. **Sanjeev Kumar Suman and L.B. Roy** served as the research supervisors, providing critical assessment of the manuscript.

References

[1] Liang S, Jianchun X, Xun Z. An extraction and classification algorithm for concrete cracks based on machine vision. IEEE Access. 2018; 6:45051-61.
[2] Munawar HS, Hammad AW, Haddad A, Soares CA, Waller ST. Image-based crack detection methods: a review. Infrastructures. 2021; 6(8):1-20.
[3] Cao W, Liu Q, He Z. Review of pavement defect detection methods. IEEE Access. 2020; 8:14531-44.
[4] Sarsam SI. Pavement maintenance management system: a review. Trends in Transport Engineering and Applications. 2016; 3(2):19-30.
[5] Kheradmandi N, Mehranfar V. A critical review and comparative study on image segmentation-based techniques for pavement crack detection. Construction and Building Materials. 2022; 321:126162.
[6] Heitzmann P. A computer vision-assisted approach to automated real-time road infrastructure management. arXiv preprint arXiv:2202.13285. 2022.
[7] Aashto BD. American association of state highway and transportation officials. Washington, DC. 2010; 4.
[8] Peraka NS, Biligiri KP. Pavement asset management systems and technologies: a review. Automation in Construction. 2020; 119:103336.

[9] Yang B, Qin Z, Zhou Q, Li H, Li L, Yang X. Pavement damage behaviour of urban roads in seasonally frozen saline ground regions. Cold Regions Science and Technology. 2020; 174:103035.
[10] Sari Y, Prakoso PB, Baskara AR. Road crack detection using support vector machine (SVM) and OTSU algorithm. In 6th international conference on electric vehicular technology 2019 (pp. 349-54). IEEE.
[11] Chun PJ, Ujike I, Mishima K, Kusumoto M, Okazaki S. Random forest-based evaluation technique for internal damage in reinforced concrete featuring multiple nondestructive testing results. Construction and Building Materials. 2020; 253:119238.
[12] Ghaemi M, Feizi-derakhshi MR. Forest optimization algorithm. Expert Systems with Applications. 2014; 41(15):6676-87.
[13] Yang P, Song W, Zhao X, Zheng R, Qingge L. An improved otsu threshold segmentation algorithm. International Journal of Computational Science and Engineering. 2020; 22(1):146-53.
[14] Sun Z, Caetano E, Pereira S, Moutinho C. Employing histogram of oriented gradient to enhance concrete crack detection performance with classification algorithm and Bayesian optimization. Engineering Failure Analysis. 2023; 150:107351.
[15] Karthikeyan A, Pavithra S, Anu PM. Detection and classification of 2D and 3D hyper spectral image using enhanced Harris corner detector. Scalable Computing: Practice and Experience. 2020; 21(1):93-100.
[16] Lubis A, Iskandar I, Panjaitan ML. Implementation of KNN methods and GLCM extraction for classification of road damage level. IAIC Transactions on Sustainable Digital Innovation. 2022; 4(1):1-7.
[17] Cheng HD, Chen JR, Glazier C, Hu YG. Novel approach to pavement cracking detection based on fuzzy set theory. Journal of Computing in Civil Engineering. 1999; 13(4):270-80.
[18] Chen C, Seo H, Jun CH, Zhao Y. Pavement crack detection and classification based on fusion feature of LBP and PCA with SVM. International Journal of Pavement Engineering. 2022; 23(9):3274-83.
[19] Islam MM, Hossain MB, Akhtar MN, Moni MA, Hasan KF. CNN based on transfer learning models using data augmentation and transformation for detection of concrete crack. Algorithms. 2022; 15(8):1-17.
[20] Hoang ND, Nguyen QL. Computer vision-based recognition of pavement crack patterns using light gradient boosting machine, deep neural network, and convolutional neural network. Journal of Soft Computing in Civil Engineering. 2023; 7(3):21-51.
[21] Li D, Duan Z, Hu X, Zhang D. Pixel-level recognition of pavement distresses based on U-Net. Advances in Materials Science and Engineering. 2021; 2021:1-11.
[22] Fan Z, Lin H, Li C, Su J, Bruno S, Loprencipe G. Use of parallel ResNet for high-performance pavement crack detection and measurement. Sustainability. 2022; 14(3):1-21.
[23] Yu Y, Samali B, Rashidi M, Mohammadi M, Nguyen TN, Zhang G. Vision-based concrete crack detection using a hybrid framework considering noise effect. Journal of Building Engineering. 2022; 61:105246.
[24] Elhariri E, El-bendary N, Taie SA. Using hybrid filter-wrapper feature selection with multi-objective improved-salp optimization for crack severity recognition. IEEE Access. 2020; 8:84290-315.
[25] Matarneh S, Elghaish F, Al-ghraibah A, Abdellatef E, Edwards DJ. An automatic image processing based on hough transform algorithm for pavement crack detection and classification. Smart and Sustainable Built Environment. 2023.
[26] Ashraf A, Sophian A, Shafie AA, Gunawan TS, Ismail NN, Bawono AA. Efficient pavement crack detection and classification using custom YOLOv7 model. Indonesian Journal of Electrical Engineering and Informatics. 2023; 11(1):119-32.
[27] Guo F, Qian Y, Liu J, Yu H. Pavement crack detection based on transformer network. Automation in Construction. 2023; 145:104646.
[28] Tello-cifuentes L, Marulanda J, Thomson P. Detection and classification of pavement damages using wavelet scattering transform, fractal dimension by box-counting method and machine learning algorithms. Road Materials and Pavement Design. 2023:1-9.
[29] Tran TS, Tran VP, Lee HJ, Flores JM, Le VP. A two-step sequential automated crack detection and severity classification process for asphalt pavements. International Journal of Pavement Engineering. 2022; 23(6):2019-33.
[30] Hammouch W, Chouiekh C, Khaissidi G, Mrabti M. Crack detection and classification in moroccan pavement using convolutional neural network. Infrastructures. 2022; 7(11):1-12.
[31] Ahmadi A, Khalesi S, Golroo A. An integrated machine learning model for automatic road crack detection and classification in urban areas. International Journal of Pavement Engineering. 2022; 23(10):3536-52.
[32] Hoang ND, Huynh TC, Tran XL, Tran VD. A novel approach for detection of pavement crack and sealed crack using image processing and salp swarm algorithm optimized machine learning. Advances in Civil Engineering. 2022; 2022:1-21.
[33] Jana S, Thangam S, Kishore A, Sai KV, Vandana S. Transfer learning based deep convolutional neural network model for pavement crack detection from images. International Journal of Nonlinear Analysis and Applications. 2022; 13(1):1209-23.
[34] Liu C, Xu B. A night pavement crack detection method based on image-to-image translation. Computer-Aided Civil and Infrastructure Engineering. 2022; 37(13):1737-53.
[35] Ali R, Chuah JH, Talip MS, Mokhtar N, Shoaib MA. Structural crack detection using deep convolutional neural networks. Automation in Construction. 2022; 133:103989.
[36] Lahmiri S. Integrating convolutional neural networks, KNN, and bayesian optimization for efficient

diagnosis of Alzheimer's disease in magnetic resonance images. Biomedical Signal Processing and Control. 2023; 80:104375.
[37] https://www.kaggle.com/datasets/lakshaymiddha/crack-segmentation-dataset. Accessed 23 January 2024.
[38] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In proceedings of the conference on computer vision and pattern recognition 2015 (pp. 3431-40). IEEE.
[39] Liu W, Huang Y, Li Y, Chen Q. FPCNet: fast pavement crack detection network based on encoder-decoder architecture. arXiv preprint arXiv:1907.02248. 2019.
[40] Chen FC, Jahanshahi MR. NB-CNN: deep learning-based crack detection using convolutional neural network and naïve bayes data fusion. IEEE Transactions on Industrial Electronics. 2017; 65(5):4392-400.
[41] Escalona U, Arce F, Zamora E, Sossa H. Fully convolutional networks for automatic pavement crack segmentation. Computación y Sistemas. 2019; 23(2):451-60.
[42] Cheng J, Xiong W, Chen W, Gu Y, Li Y. Pixel-level crack detection using U-Net. In TENCON region 10 conference 2018 (pp. 462-6). IEEE.
[43] Oliveira H, Correia PL. Automatic road crack detection and characterization. IEEE Transactions on Intelligent Transportation Systems. 2012; 14(1):155-68.
[44] Shi Y, Cui L, Qi Z, Meng F, Chen Z. Automatic road crack detection using random structured forests. IEEE Transactions on Intelligent Transportation Systems. 2016; 17(12):3434-45.
[45] Liu Y, Cheng MM, Hu X, Wang K, Bai X. Richer convolutional features for edge detection. In proceedings of the conference on computer vision and pattern recognition 2017 (pp. 3000-9). IEEE.
[46] Anand S, Gupta S, Darbari V, Kohli S. Crack-pot: autonomous road crack and pothole detection. In digital image computing: techniques and applications 2018 (pp. 1-6). IEEE.

Shivangi Mishra is a Research Scholar in the department of Civil Engineering at NIT Patna, Bihar, India. She has more seven years teaching and research experience. She holds a Mtech degree in Geoinformatics and its applications. Her research areas are in the area of Remote Sensing, GIS and Application of soft computing in the field of civil engineering. She has published five research papers in international journals and conference proceedings.
Email: shivangicivil0038@gmail.com

Dr Sanjeev Kumar Suman is currently working as an Associate Professor in the department of Civil Engineering at NIT Patna, Bihar, India. He has eighteen years teaching and research experience. He hold a PhD degree in transportation engineering. His research interests include sustainable and waste pavement materials, NDT pavement evaluation, Pavement management system, Construction quality control and assurance etc. He has published more than 48 papers in international journals of repute and conferences proceedings. He has also handled industrial consultancy projects and guided multiple research scholars.
Email: sksuman@nitp.ac.in

Dr L.B. Roy is working as a Professor of Civil Engineering at National Institute of Technology Patna, India since August 2013. He has worked as Dy. Registrar (Examination Service) for six years (2006-2011), as Head, Civil Engineering Department for two years (2011-13), Associate Dean Academics for one semester (July 2013 – December 2013), as Dean Academics for two years (2014-2016), and Dean (Research & Consultancy) for three years (2019-2022). Prior to this, he has worked as an Associate Professor of Civil Engineering at NIT Patna for seven years (2006-2013), as an Associate Professor for 8.5 years (i.e. from 1995 to 2001 and from 2002 to 2004), as an Dean Research and Consultancy for 3 years (i.e. from April-2019 to April-2022), and Head of Irrigation Engineering Department (4 years) at Arba Minch University in Ethopia. As an Assistant Professor at Water and Land Management Institute Patna for ten years (from 1988 to 1995, 2001 to 2002 and 2004 to 2006), as an Assistant Engineer for one year in Water Recourses Department Govt. of Bihar (i.e., 1987-1988). He has altogether 38 years' experience in teaching, training, research, and academic administration. His specialization includes irrigation water management with special focus on land drainage, soil stabilization and swelling clays, Soil Mechanics and Foundation Engineering, Rock Mechanics and Tunnel Engineering. Dr Roy has presented technical papers at several national and international seminars and conferences in India and other countries. More than forty national and international journal papers are to his credit. He has successfully guided 14 Ph.D. Thesis, in the specialization of Water Resources and Geotechnical Engineering. He also worked as member of the Editorial Board of the Ethiopian Journal of Water Science and Technology during the period (1996-2004). He is a regular reviewer for reputed Scopus indexed and Web of Science indexed journals, namely the Journal of Engineering, Technology and Applied Science Research (ETASR) and the Journal of Agricultural Research Communication Centre. He is a Master Trainer on Irrigation Water Management, and he has undergone several short- and long-term trainings in the field of IWM and Geotechnical

Engineering, which includes the six months long term Training of Trainers, i. e. 8th TOT at WALMI Aurangabad and field visits in USA, Spain and México. He is also a widely travelled person, e. g. USA, Mexico, Spain, Ethiopia, Iran, Kuwait, Indonesia, Malaysia, China, Yemen, Kenya, Nepal and Australia. He has also authored of two textbooks, with titles ‘Application of Graphic in Engineering and Static Graphics, respectively. His degree includes Ph. D. (1995) from Patna University (NIT Patna), M. E. Horn’s (1987) from University of Roorkee (now IIT Roorkee) and B.Sc. (Engg) with Distinction from RIT Jamshedpur under Ranchi University (now NIT Jamshedpur), India.
Email: lbroy@nitp.ac.in

Appendix I

S. No.	Abbreviation	Description
1	AIU	Average Improvement of Utilities
2	ANN	Artificial Neural Network
3	ASINVOSnet	Assisting and Interactive Machine Learning Based Visual Monitoring System Network
4	BO-kNN	Bayesian-Optimized k-Nearest Neighbors Classifier
5	CFD	Crack-Forest Dataset
6	CNN	Convolutional Neural Networks
7	CT	Crack Transformer
8	FCN	Fully Convolutional Network
9	FOA	Forest Optimization Algorithm
10	FPCNet	Fully Point-Wise Convolutional Neural Network
11	GLCM	Gray Level Co-Occurrence Matrix
12	HED	Holistically-Nested Edge Detection
13	HOG	Histograms of Oriented Gradients
14	kNN	*k*-Nearest Neighbors
15	MCC	Matthews Correlation Coefficient
16	MLP	Multilayer Perceptron
17	NB-CNN	Naïve Bayes Based Convolutional Neural Network
18	ODS	Object Detection Score
19	OIS	Object Identification Score
20	RCDnet	Rain Convolutional Dictionary Network
21	RCF	Richer Convolutional Features
22	RCNN	Region Based Convolutional Neural Networks
23	RF	Random Forest
24	SVM	Support Vector Machines
25	VGG-19	Visual Geometry Group-19
26	VGGNet	Visual Geometry Group Network
27	YOLOv7	You Only Look Once