

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/339003087>

# Design of a new low-cost unmanned aerial vehicle and vision-based concrete crack inspection method

Article in *Structural Health Monitoring* · February 2020

DOI: 10.1177/1475921719898862

CITATIONS

0

READS

134

5 authors, including:



Ning Wang

University of Houston

50 PUBLICATIONS 278 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Virtual and Remote Laboratory (VR-Lab ) Development [View project](#)



Integration and Optimization of Building Information Modeling and Electronic Commerce Material Procurement Process [View project](#)

# Design of a new low-cost unmanned aerial vehicle and vision-based concrete crack inspection method

Structural Health Monitoring

1–13

© The Author(s) 2020

Article reuse guidelines:

sagepub.com/journals-permissions

DOI: 10.1177/1475921719898862

journals.sagepub.com/home/shm



Bin Lei<sup>1</sup>, Yali Ren<sup>2</sup>, Ning Wang<sup>3</sup>, Linsheng Huo<sup>4</sup> and Gangbing Song<sup>3</sup>

## Abstract

With the explosive development of the computer vision technology, more and more vision-based inspection methods enabled by unmanned aerial vehicle technologies have been researched on the crack inspection of the sundry concrete structures. However, because of the limitation of the low-cost unmanned aerial vehicle hardware, whose cost is around US\$500, most of the vision-based methods are difficult to be implemented on the low-cost unmanned aerial vehicle for real-time crack inspection. To address this challenge, in this article, a new computationally efficient vision-based crack inspection method is designed and successfully implemented on a low-cost unmanned aerial vehicle. Furthermore, to reduce the acquired data samples, a new algorithm entitled crack central point method is designed to extract the effective information from the pre-processed images. The proposed vision-based crack detection method includes the following three major components: (1) the image pre-processing algorithm, (2) crack central point method, and (3) the support vector machine model-based classifier. To demonstrate the effectiveness of the new inspection method, a concrete structure inspection experiment is implemented. The experimental results indicate that this new method is able to accurately and rapidly inspect the cracks of concrete structure in real time. This new vision-based crack inspection method shows great promise for the practical application.

## Keywords

Vision-based inspection, crack inspection, computer vision, support vector machine, structural health monitoring, unmanned aerial vehicle, crack central point method

## Introduction

Aging and degradation of civil and industrial infrastructures has posed a notable public safety threat.<sup>1,2</sup> In civil infrastructures, a major type of concrete damage is concrete crack, which is caused by a range of complex reasons,<sup>3,4</sup> and is an important factor in impairing structural safety.<sup>5</sup> As the structure becomes more complicated, the research on the advanced solutions to the crack inspection and detection has attracted significant attention.<sup>5–7</sup> Consequently, an increasingly number of structural health monitoring (SHM) applications with noncontact means have been developed and applied for the monitoring and inspection of cracks in a variety of critical concrete structures,<sup>8,9</sup> such as plates,<sup>10</sup> columns,<sup>11,12</sup> beams,<sup>13</sup> tubes,<sup>14,15</sup> bridges,<sup>16–20</sup> dam,<sup>21</sup> wind turbine,<sup>22,23</sup> and others. Compared to the traditional contact methods, the majority of noncontact SHM methods have improvement in the convenience and efficiency of structural inspection and crack detection.<sup>13</sup>

However, noncontact SHM methods can still be complex,<sup>24</sup> time-consuming,<sup>25,26</sup> labor-intensive,<sup>16,19,26</sup> expensive,<sup>21,24,25</sup> or error-prone due to the limitation of the skill and experience of inspectors.<sup>16</sup> In addition, it engenders unexpected safety risks when the inspectors

<sup>1</sup>School of Mechanical Engineering and Automation, Wuhan University of Science and Technology, Wuhan, China

<sup>2</sup>School of Computer Science, College of Computing, Georgia Institute of Technology, Atlanta, GA, USA

<sup>3</sup>Department of Electrical and Computer Engineering, University of Houston, Houston, TX, USA

<sup>4</sup>School of Civil and Hydraulic Engineering, Dalian University of Technology, Dalian, China

## Corresponding author:

Gangbing Song, Department of Mechanical Engineering, University of Houston, Houston, TX 77204, USA.

Email: gsong@uh.edu

access the unsafe structural areas.<sup>21,24,25</sup> Therefore, developing the cost-effective, accurate, and automated noncontact solutions for efficient and reliable SHM has been an emerging research topic.<sup>19</sup>

Nowadays, the rapid development of unmanned aerial vehicle (UAV) propels the SHM of large-scale concrete structures to advance in the direction of process automation and remote monitoring.<sup>27</sup> Currently, UAVs can carry a wide range of imagery and sensing instruments, including cameras,<sup>25</sup> radars,<sup>28</sup> optical imagery,<sup>28,29</sup> infrared thermography,<sup>28</sup> light detection and ranging (LIDAR) sensors,<sup>29</sup> microwave systems,<sup>28</sup> and so on. Moreover, UAVs can have small size,<sup>28</sup> low-cost,<sup>28</sup> excellent maneuverability and versatility,<sup>30</sup> and adaptability to complex environments.<sup>31</sup> Therefore, UAVs can perform the inspections of structures which are inconveniently accessible by traditional inspectors.<sup>28</sup> Especially, UAVs integrated with optic devices and image processing have been increasingly applied in the photogrammetric three-dimensional (3D) structural model reconstruction and analysis,<sup>21,28</sup> remote sensing,<sup>21,32</sup> and various structural health detections.<sup>23,28–30,33,34</sup> Generally, the application of UAVs can overcome many flaws of traditional SHM methods;<sup>25,35</sup> improve the quality of inspections;<sup>24</sup> reduce inspection time,<sup>19</sup> cost and work incidents;<sup>33</sup> decrease logistics<sup>33</sup> and traffic interference;<sup>19</sup> and increase structural accessibility.<sup>19,28</sup>

Computer vision technology applies multi-disciplinary technologies—which include optical imaging, image processing, artificial intelligent, and machine learning—to derive meaningful information and gain high-level understanding from images or videos.<sup>36</sup> In the last two decades, computer vision has been increasingly utilized in SHM<sup>17,18,26,33,37</sup> which expedites inspection process, decreases costs and safety risk, increases accuracy, and facilitates the access of structure.<sup>28</sup> Unquestionably, computer vision-based SHM (CVSHM) has been a state-of-the-art research area.<sup>28,37</sup> Recently, computer vision-based and UAV-enabled methods have been implemented for SHM of sundry concrete structures, which is shown in Table 1. However, almost all applications in Table 1 just use UAVs as a data collection tool for SHM. Even though Pereira et al.<sup>31</sup> and Sankarasrinivasan et al.<sup>34</sup> have proposed the integrated UAV and image processing system to enable real-time SHM, the integration of UAVs and computer vision technology has not been fully explored and implemented to automate and optimize the SHM process. Till now, how to integrate computer vision with the cutting-edge machine learning algorithm into low-cost UAVs for accurate, efficient, and real-time SHM remains an unsolved issue.

To address this issue, this article proposes a vision-based crack inspection method with support vector machine (SVM), which can be integrated into the low-cost UAV. To the best knowledge of the authors, this is the groundbreaking research to develop a low-cost UAV-enabled, vision and machine learning-based, real-time crack inspection method with the following two significant contributions:

1. To acquire the smaller set of valid data samples for the further crack recognition, a new crack central point method (CCPM)<sup>41</sup> has been developed to extract the crack characters from the pre-processed images. CCPM improved adaptability and robustness compared with both the traditional edge detection methods—such as Canny, Prewitt, and Sobel algorithms—and *k*-means clustering-based method. With CCPM, the size of training data set for the crack recognition can be significantly reduced,<sup>41</sup> which provides the possibility of fast computation and real-time crack detection.
2. To recognize different types of concrete cracks accurately and rapidly with fewer training samples, an SVM-based crack classification algorithm has been designed and implemented. As our CCPM algorithm<sup>41</sup> can reduce the training data significantly through extracting the crack characters from the pre-processed images, the model training and the crack feature classification time using SVM are both reduced significantly. Hence, due to its low computational complexity and high detection accuracy, this algorithm can be implemented in the low-cost UAVs for real-time and automated crack inspection.

The rest of this article is organized as follows: Section “A new vision-based crack inspection method” presents the design process and mathematical models of the proposed vision-based crack inspection method. Section “Experiments and result analyses” describes the experiment to illustrate the effectiveness and applicability of the novel vision-based method. Concluding remarks and future works are summarized in section “Conclusion and future work.”

## A new vision-based crack inspection method

The goal of our research is to develop an effective recognition method to quickly inspect and detect various types of cracks from images captured by the UAVs. Meanwhile, this proposed method should have low computational complexity to enable the real-time crack inspections using the low-cost UAV devices. To design

**Table 1.** The examples of UAV vision-based SHM application.

Authors	Applications	Data collection	Image processing methods
Khaloo et al. <sup>21</sup>	Applied multiple UAVs and multi-scale photogrammetry for dam inspection.	Collected the images by the UAVs from a hydroelectric dam.	Multi-scale photogrammetry and computer vision to detect dam defects.
Hallermann et al. <sup>24</sup>	Applied the low-cost UAVs to monitor large-scale structures (dams, walls, etc.)	Collected the images by UAVs.	Computer vision to detect displacement and deformation in postflight analysis.
Kim et al. <sup>25</sup>	Applied UAVs for the assessment of concrete cracks.	Collected the images by the UAVs.	Image binarization and hybrid image processing in workstations.
Brooks et al. <sup>29</sup>	Applied UAVs for traffic monitoring and infrastructural health inspection.	Collected the images and other information by the UAVs.	Vision-based image processing to reconstruct 3D sites and identify damage.
Pereira et al. <sup>31</sup>	Applied UAVs to automatically detect cracks in civil constructions.	Captured the images and other data by the UAVs.	Edge detection and particle filter in MATLAB/embedded sensors on UAVs.
Ellenberg et al. <sup>32</sup>	Applied UAVs for remote sensing and inspection of the large-scale infrastructure condition.	Collected the high-quality images and other data by UAVs.	Vision-based image processing and quantitative assessment to evaluate the infrastructural health.
Sankarasrinivasan et al. <sup>34</sup>	Applied UAVs to detect crack and surface degradation of large structures.	Acquired the images and other data by the UAVs.	Hat transforms, HSV, and grayscale thresholding in MATLAB.
Wang and Zhang <sup>38</sup>	Applied UAVs to inspect the cracks of wind turbine blade surface.	Collected the high-quality images by the UAVs.	Haar-like features, such as LogitBoost, Decision Tree, and SVM to detect cracks.
Eschmann et al. <sup>39</sup>	Applied UAVs for remote damage assessment and monitoring purposes.	Captured the high-resolution images by the UAVs.	Image post-processing and pattern recognition to assess damages.
Kang and Cha <sup>40</sup>	Applied UAVs using ultrasonic beacons for damage localization and detection.	Collected the images by the UAVs.	Deep CNN to detect damage.

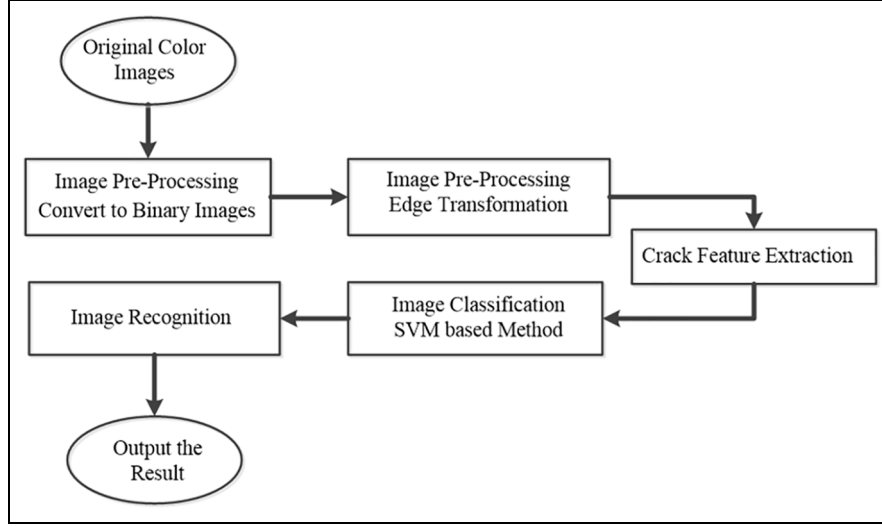
UAV: unmanned aerial vehicle; HSV: hue, saturation, and value; SHM: structural health monitoring; SVM: support vector machine; CNN: convolutional neural network.

this method, the whole process begins from the principle analysis of the research problem. Then, mathematical models of each step are assembled to ensure the validity of key design components of this method.

### Principle statement

Traditionally, a computer vision task consists of the methods of obtaining, processing, analyzing, and interpreting digital images or videos, and extracting multi-dimensional data to generate numerical or symbolic information.<sup>36</sup> To rapidly and effectively detect the cracks from the captured images, an innovative vision-based and SVM-supported crack inspection method is proposed. Figure 1 shows the working process of the proposed inspection method, which includes the following three steps: (1) original image pre-processing, (2) crack feature extraction with CCPM method, and (3) crack classification and recognition. The general illustration of each step is presented as follows.

First, the images of the concrete surface, which are acquired by the UAVs, are converted to grayscale binary images via pre-processing. Compared with the color images, images converted to grayscale can greatly reduce the computational load of data processing. Meanwhile, the converted grayscale images have no negative impact on crack detection. Second, CCPM is applied to extract the center of each crack from the grayscale binary images. CCPM has the improved performance of the traditional edge detection methods, especially when the images are captured with uneven natural light, dirty spots, and rusts on the structural surface, and environment noise.<sup>41</sup> In addition, CCPM has advantages over  $k$ -means clustering algorithms, due to the more computational time, sensitivity to noise points, and variability of  $k$  value in  $k$ -means clustering.<sup>41</sup> In this crack extraction process, the crack is characterized by the existing minimum grayscale value, which exhibits the property of parabolic function in the row of crack area. The eigenvalues selected by CCPM



**Figure 1.** Working process of proposed method.

can differentiate four types of cracks: transverse, longitudinal, oblique, and mesh.<sup>41</sup> Finally, the characteristics of the extracted cracks should be mutually independent. To meet the real-time computational capacity of the low-cost UAV, the SVM-based method is selected as the crack classifier. The crack projection characteristics in  $X$ - and  $Y$ -coordinates, crack area, and distributed density are the input features.

### Mathematical model for proposed method

According to the working process in Figure 1, the corresponding mathematical model of each step is shown as follows:

1. Original image pre-processing: the original digital color image  $I_o$  is represented by an  $H_o \times W_o \times 3$  matrix, where  $H_o$  and  $W_o$  are the height and width of the image, respectively. The pixel  $x_{i,j}$  in image  $I_o$  is denoted as  $[b, g, r]$  representing the color of the pixel, where  $i \in \{0, 1, \dots, H_o\}$  and  $j \in \{0, 1, \dots, W_o\}$ , respectively.  $I_o$  is then converted into the binary image  $I_b$  in grayscale. The threshold method is applied to create  $I_b$ . To reduce the impact of the different lighting conditions on the image, adaptive threshold method is used. The value in pixel  $x_{i,j}$  in  $I_b$  is determined by

$$x_{i,j} = \begin{cases} \text{MaxValue}, & \text{if } x_{i,j} \text{ in } I_b > W_{i,j} \\ 0, & \text{otherwise} \end{cases}$$

where  $W_{i,j}$  is the weighted sum (cross-correlation with a Gaussian window) of the neighborhood of  $x_{i,j}$ .

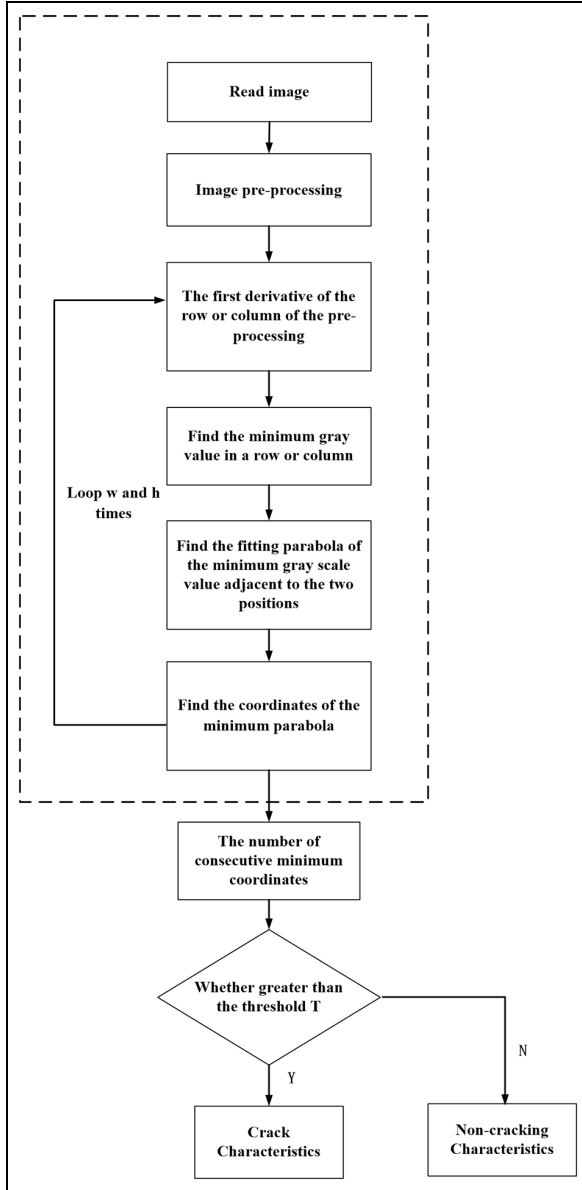
The contours  $C$  in the binary grayscale image are detected based on the topological analysis of binary images introduced in Suzuki et al.,<sup>42</sup> in which the contours are determined to be the boundaries between the black and white points. Each detected contour of  $C$  is a set of two-dimensional (2D) points and  $i \in \{1, 2, \dots, N\}$ , where  $N$  is the number of contours. The ellipse fitting of the detected contours  $C$  is based on the least squares algorithm.<sup>43</sup> In our method, given a set of 2D points in the image  $P = \{p_i\}_{i=1}^N$ , where  $p_i = (x_i, y_i)$  and  $x_i, y_i$  are the two-dimensional coordinates of the point,  $p_i$ . An ellipse curve  $C(\alpha)$  is parameterized by the vector  $\alpha$  and the distance metric  $\delta(C(\alpha), p_i)$ , which measures the distance from a point  $p_i$  to the curve  $C(\alpha)$ ; the global minimum grayscale value through the objective function is given by

$$L(p_i) = \arg \min_{P = \{p_i\}_{i=1}^N} \sum_{i=1}^N \sum_{p_i \in P} \delta(C(\alpha), p_i) \quad (1)$$

where  $C(\alpha) = p_i \cdot \alpha$  and  $\delta(C(\alpha), p_i) = \|C(\alpha) - p_i\|^2$ . Then, the minimization problem is to minimize the objective function

$$L(p_i) = \arg \min_{P = \{p_i\}_{i=1}^N} \sum_{i=1}^N \sum_{p_i \in P} \|C(\alpha) - p_i\|^2 = \|D\alpha\|^2 \quad (2)$$

which is subject to the constraint that  $\|\alpha\|^2 = 1$ . The design matrix  $D$  is an  $n \times 6$  matrix with rows  $x_i$ . Then, the problem becomes an eigenvector problem  $D^T D \alpha = \lambda \alpha$ , where  $\alpha_{min}$  is the eigenvector corresponding to the smallest eigenvalue.



**Figure 2.** Crack inspection process of CCPM method.<sup>34</sup>

- Crack feature extraction using CCPM method: the crack is characterized based on the existed minimum grayscale value which displays a parabolic property and is calculated from equation (2). Figure 2 illustrates the whole crack inspection process of CCPM method.<sup>41</sup> Through analyzing the characteristics of the crack areas in the pre-processing image, the image function is  $f(r, c)$ . The gradient of the points in digital image is a vector. Then, the gradient is determined by the vector of the first-order partial derivative

$$\Delta f(r, c) = \left( \frac{\partial f(r, c)}{\partial r}, \frac{\partial f(r, c)}{\partial c} \right) = (f_r, f_c) \quad (3)$$

where  $f_r$  and  $f_c$  are the gradients along the row and column directions, respectively. The respective row and column derivatives, and direction angles are

$$\begin{cases} (\Delta f(r), \Delta f(c)) = \left( \frac{\partial f(r, c)}{\partial r}, \frac{\partial f(r, c)}{\partial c} \right) \\ \phi = -\arctan\left(\frac{f_r}{f_c}\right) \\ f'_i = \frac{1}{2}(f_{i+1} - f_{i-1}) \end{cases} \quad (4)$$

where  $f'_i = 0$ ,  $i \in \{1, 2, \dots, N\}$ , and the minimum grayscale value can be obtained at the corresponding point. Since the distribution of the grayscale value  $f(r, c)$  in crack area is a parabolic function, two adjacent points  $r$  and  $c$  can be taken to fit the parabola with  $f(r-1, c-1), f(r+1, c+1)$ . Then, the symmetrical line position of the parabola, the center of the crack  $(r_i, c_j)$ , where  $i, j \in \{1, 2, \dots, N\}$  in a row or column, can be calculated.

However, in the real complex environment, some factors—including the presence of noise, rust, and others—result in the grayscale value to be no more than the minimum grayscale value in the crack area.<sup>41</sup> Consequently, the results of the crack detection, feature extraction, and crack classification are affected as well. To address this essential issue, the threshold can be adjusted as a basic characteristic of detecting the crack according to the features of the crack in the image. The threshold is  $T$ , and the detected crack width is  $W$ . The characteristics are

$$\begin{cases} W \geq T, & (\text{crack characteristics}) \\ 0 \leq W < T, & (\text{other characteristics}) \end{cases}$$

The crack width is defined to be larger than 2 mm.<sup>44</sup> If the length of the recognized feature is no less than the threshold (100 or 400), it is considered as a crack. Otherwise, it is not.

To find the edge coordinates of the crack, we take the second derivative of equation (3)

$$\Delta^2 f(r, c) = \left( \frac{\partial^2 f(r, c)}{\partial r^2} + \frac{\partial^2 f(r, c)}{\partial c^2} \right) = (f_{rr} + f_{cc}) \quad (5)$$

In practice, the formula of second derivatives is

$$f''(i) = \frac{1}{2}(f_{i+1} - 2f_i + f_{i-1})$$

When  $f''_i = 0$ , we can obtain its maximum and minimum. The crack edge coordinates are the coordinates of maximum and minimum values,  $(r_{i+1}, c_{j+1})$  and  $(r_{i-1}, c_{j-1})$ , respectively.

Finally, the crack width,  $W$ , can be calculated as follows

$$W = 2 \times \left\{ \frac{RowR - Row}{\cos(\phi)} \right\} \quad (6)$$

where  $RowR$  and  $Row$  are the crack right and center position, respectively.

3. Crack classification and recognition using the SVM model: to develop an effective and accurate crack classifier for the low-cost UAV equipment, based on our preliminary research, SVM model<sup>45</sup> is a good selection. The key ideas and advantages of SVM include the following: (1) the efficient separability of nonlinear regions, which use the kernel functions to generalize the similarity to the new similarity measures by dot products;<sup>46</sup> (2) the use of quadratic optimization to avert local minimized issues in artificial neural network;<sup>46</sup> and (3) the result learning algorithm will be an optimization algorithm instead of a greedy search.<sup>46</sup> Thus, compared to the other traditional machine learning or pattern recognition methods, SVM requires lesser data samples.<sup>45</sup> In our proposed method, the sequential minimal optimization (SMO) algorithm is applied to deal with the image classification and recognition. SMO algorithm was developed to solve the SVM quadratic programming (QP) problem.<sup>47</sup> SMO algorithm first decomposes the SVM QP problem into several QP sub-problems<sup>47</sup> and then solves each smallest possible optimization problem individually,<sup>47</sup> which involves two Lagrange multipliers at each step.

SVM model includes two different forms: linear and nonlinear form. Considering the complexity of the available data sets and application, nonlinear form SVM model is selected for the classification. The output of SVM is explicitly computed from the following formula

$$f(\vec{x}) = \sum_{i=1}^N y_i \alpha_i K(\vec{x}_i, \vec{x}) - b, \quad i \in \{1, 2, \dots, N\} \quad (7)$$

where  $y_i$  is the label set,  $\alpha_i$  is the Lagrange multipliers,  $b$  is the bias, and  $K$  is the kernel function that measures the similarity or distance between the input vector  $\vec{x}$  and the stored training vector  $\vec{x}_i$ . In the presented method, Gaussian kernel is finally selected due to its higher efficiency, lower simplicity, and better performance compared to other SVM kernels. The Gaussian kernel function is shown as follows

$$K(\vec{x}_i, \vec{x}) = \exp\left(-\frac{\|\vec{x} - \vec{x}_i\|^2}{2\sigma^2}\right) \quad (8)$$

where  $\sigma$  is a free parameter, and it is utilized to control the underfitting (large  $\sigma^2$  for high bias and low

variance) and overfitting (small  $\sigma^2$  for low bias and high variance).

The Lagrange multiplier  $\alpha_i$  is computed via the quadratic program. The nonlinearity alters the quadratic form, but the dual objective function  $\Psi$  remains quadratic in  $\alpha$ . Thus, the object function of SVM model is given as

$$\begin{aligned} \min_{\vec{\alpha}} \Psi(\vec{\alpha}) &= \min_{\vec{\alpha}} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N y_i y_j K(\vec{x}_i, \vec{x}_j) \alpha_i \alpha_j - \sum_{i=1}^N \alpha_i \\ 0 &\leq \alpha_i \leq C, \quad \forall i \\ \sum_{i=1}^N y_i \alpha_i &= 0 \end{aligned} \quad (9)$$

where  $C$  is the correction parameter.

Furthermore, Karush–Kuhn–Tucker (KKT) conditions are necessary and sufficient for an optimal point of a positive definite SVM QP problem,<sup>48</sup> and  $K$  must obey Mercer's conditions.<sup>49</sup> KKT conditions for the QP problem in equation (9) are particularly simple. The QP problem is solved below

$$\begin{cases} \alpha_i = 0 \Leftrightarrow y_i f_i(\vec{x}_i) \geq 1 \\ 0 < \alpha_i < C \Leftrightarrow y_i f_i(\vec{x}_i) = 1 \\ \alpha_i = C \Leftrightarrow y_i f_i(\vec{x}_i) \leq 1 \end{cases} \quad (10)$$

where  $i \in \{1, 2, \dots, N\}$ , and  $f_i(\vec{x}_i)$  is the output of the SVM obtained from equation (7) for the  $i$ th input training sample vector  $\vec{x}_i$ .

To solve the SVM QP problem, SMO is used to optimize the objective function, equation (9) based on KKT conditions as equation (10). SMO can quickly solve the SVM QP problem without any extra matrix storage or using numerical QP optimization steps at all. SMO decomposes the overall QP problem into QP sub-problems, using Osuna's theorem to ensure convergence. The SMO algorithm includes the following three steps: (1) an analytic method for solving for the two Lagrange multipliers, (2) a heuristic for selecting multipliers to optimize, and (3) computing the threshold. The three steps are shown as follows:

Step 1: solving for two Lagrange multipliers.

In SMO, all quantities referring to the first multiplier have a subscript 1, and a subscript 2 for the second multiplier. Without loss of generality, the algorithm first computes the second Lagrange multiplier  $\alpha_2$  and computes the ends of the diagonal line segment in terms of  $\alpha_2$ . If the target  $y_1$  does not equal the target  $y_2$ , then the following bounds apply to  $\alpha_2$

$$L = \max(0, \alpha_2 - \alpha_1), \quad H = \min(C, C + \alpha_2 - \alpha_1) \quad (11)$$

If the target  $y_1$  equals  $y_2$ , then the following bounds apply

$$L = \max(0, \alpha_2 + \alpha_1 - C), \quad H = \min(C, \alpha_2 + \alpha_1) \quad (12)$$

where  $L$  is the bound negative support vectors, and  $H$  is the bound positive support vectors.

The second derivative of the objective function along the diagonal line can be expressed as

$$\eta(\vec{x}_1, \vec{x}_2) = K(\vec{x}_1, \vec{x}_1) + K(\vec{x}_2, \vec{x}_2) - 2K(\vec{x}_1, \vec{x}_2)$$

In normal circumstances, the objective function is positive definite. In addition, there is a minimum along the direction of the linear equality constraint, and  $\eta(\vec{x}_1, \vec{x}_2)$  is greater than zero. SMO computes the minimum along the direction of the constraint

$$\alpha_2^{new} = \alpha_2 + \frac{y_2(E_1 - E_2)}{\eta(\vec{x}_1, \vec{x}_2)} \quad (13)$$

where  $E_i = f_i(\vec{x}_i) - y_i$  is the error on the  $i$ th training sample. In the next step, the constrained minimum is found by clipping the unconstrained minimum to the ends of the line segment

$$\alpha_2^{new, clipped} = \begin{cases} H & \text{if } \alpha_2^{new} \geq H \\ \alpha_2^{new} & \text{if } L < \alpha_2^{new} < H \\ L & \text{if } \alpha_2^{new} \leq L \end{cases}$$

Let  $s = y_1 y_2$ . The value of  $\alpha_1$  is computed from the newly clipped  $\alpha_2$

$$\alpha_1^{new} = \alpha_1 + s(\alpha_2 - \alpha_2^{new, clipped}) \quad (14)$$

In unusual circumstances,  $\eta$  is not positive. A negative  $\eta$  occurs if  $K$  does not obey Mercer's condition, which can cause the objective function to become indefinite. A zero  $\eta$  can occur even with a correct kernel, if more than one training sample has the same input vector  $x_i$ . In any event, SMO will work even when  $\eta$  is not positive. In those cases, the objective function  $\Psi$  should be evaluated at each end of the line segment

$$\begin{cases} f_1 = y_1(E_1 + b) + \alpha_1 K(\vec{x}_1, \vec{x}_1) - \alpha_2 K(\vec{x}_1, \vec{x}_2) \\ f_2 = y_2(E_2 + b) + \alpha_1 K(\vec{x}_1, \vec{x}_2) - \alpha_2 K(\vec{x}_2, \vec{x}_2) \\ L_1 = \alpha_1 + s(\alpha_2 - L) \\ H_1 = \alpha_1 + s(\alpha_2 - H) \\ \Psi_L = L_1 f_1 + L_2 f_2 + \frac{1}{2} L_1^2 K(\vec{x}_1, \vec{x}_1) \\ \quad + \frac{1}{2} L_2^2 K(\vec{x}_2, \vec{x}_2) + s L_1 L_2 K(\vec{x}_1, \vec{x}_2) \\ \Psi_H = H_1 f_1 + H_2 f_2 + \frac{1}{2} H_1^2 K(\vec{x}_1, \vec{x}_1) \\ \quad + \frac{1}{2} H_2^2 K(\vec{x}_2, \vec{x}_2) + s H_1 H_2 K(\vec{x}_1, \vec{x}_2) \end{cases} \quad (15)$$

SMO will move the Lagrange multipliers to the end point that has the lowest value of the objective function.

Step 2: heuristics of selecting multipliers for optimization.

To choose the multipliers for optimization, two separate choice heuristics are required, which include one for the first Lagrange multiplier and one for the second Lagrange multiplier. The choice of the first heuristic provides the outer loop of SMO. The outer loop first iterates over the entire training set, determining if each example violates the KKT conditions in equation (10). If an example violates KKT, then it is eligible for optimization. The first choice heuristic concentrates the CPU time on the examples that are most likely to violate the KKT conditions: the non-bound subset. After one passes through the entire training set, the outer loop iterates over all examples whose Lagrange multipliers are neither 0 nor  $C$  (the non-bound examples).

Again, each example is checked against the KKT conditions and violating examples are eligible for optimization. The outer loop repeatedly passes over the non-bound examples until all of the non-bound examples obey the KKT conditions within  $\varepsilon$ , which is the control parameter. Notice that the KKT conditions are checked to be within  $\varepsilon$  of fulfillment. Typically,  $\varepsilon$  is set to be  $10^{-3}$ . The outer loop then goes back and iterates over the entire training set. The outer loop keeps alternating between single pass over the entire training set and multiple passes over the non-bound subset until the entire training set obeys the KKT conditions within  $\varepsilon$  where upon the algorithm terminates.

Once the first Lagrange multiplier is chosen, the second Lagrange multiplier is chosen by SMO to maximize the step size taken during joint optimization. However, evaluating the kernel function  $K$  is time-consuming, so SMO approximates the step size by the absolute value of the numerator in equation (13):  $|E_1 - E_2|$ . SMO keeps a cached error value  $E$  for each non-bound training example and then chooses an error to approximately maximize the step size. If  $E_1$  is positive, SMO chooses an example with minimum error  $E_2$ . Conversely, if  $E_1$  is negative, SMO chooses one with maximum error  $E_2$ .

Step 3: computing the threshold.

The threshold  $b$  is recomputed after each step, so that the KKT conditions are fulfilled for both optimized examples. The following threshold  $b_1$  is valid when the new  $\alpha_1$  is not at the bounds, since it forces the output of the SVM model to be  $y_1$  when the input is  $x_1$



$$b_1 = E_1 + y_1(\alpha_1^{new} - \alpha_1)K(\vec{x}_1, \vec{x}_1) + y_2(\alpha_2^{new, clipped} - \alpha_2)K(\vec{x}_1, \vec{x}_2) + b \quad (16)$$

The following threshold  $b_2$  is valid when the new  $\alpha_2$  is not at bounds, because it forces the output of the SVM model to be  $y_2$  when the input is  $x_2$

$$b_2 = E_2 + y_1(\alpha_1^{new} - \alpha_1)K(\vec{x}_1, \vec{x}_1) + y_2(\alpha_2^{new, clipped} - \alpha_2)K(\vec{x}_1, \vec{x}_2) + b \quad (17)$$

When both  $b_1$  and  $b_2$  are valid, they are equal. When both new Lagrange multipliers are at bound and if  $L$  is not equal to  $H$ , then the interval between  $b_1$  and  $b_2$  is all thresholds that are consistent with the KKT conditions. SMO chooses the threshold to be halfway in between  $b_1$  and  $b_2$ .

## Experiments and result analyses

To verify the performance of the new vision-based crack inspection method, a bridge structure inspection experiment using the UAV combined with computer vision and SVM algorithm for crack detection is implemented, as shown in the following section.

### Hardware configuration for crack data acquisition

The hardware platform of a low-cost quadrotor UAV, whose cost is around US\$500, has 10 modules: control module, electronic spark control (ESC) module, ultrasonic module, inertial navigation module, transmission and reception module, optical flow module, communication module, global positioning system (GPS) module, power loss memory module, and the power supply module.

Control module is composed of STM32F103ZET microcomputer. It is used to control the attitude of the quadrotor UAV, exchange data between the quadrotor UAV and terminal devices, and handle the data acquisition and processing. ESC module is used to handle the driving operation of the brushless motor in the quadrotor UAV. Inertial navigation module is composed of an accelerometer, gyroscope, electronic compass, and barometer. It is used to provide the information feedback for the smooth flight of the quadrotor UAV. Ultrasonic module is used to measure the flight height of the quadrotor UAV. Transmission and reception module is used for the wireless real-time video transmission. Optical flow module is used for the fixed-point control of the quadrotor UAV. Communication module uses NRF24L01 module for the quadrotor UAV and PC side of the data transmission. GPS module is used for positioning and navigation of the quadrotor UAV in the high altitude. Power loss memory

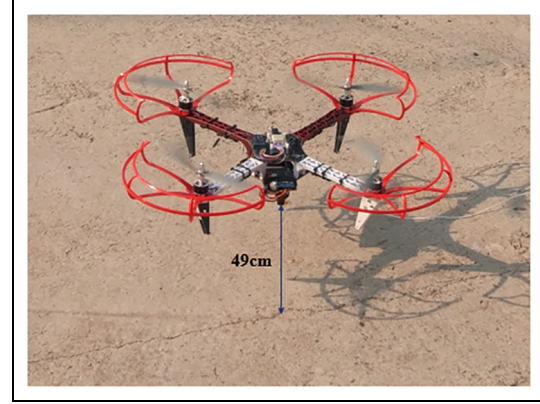


Figure 3. Hardware of low-cost UAV.

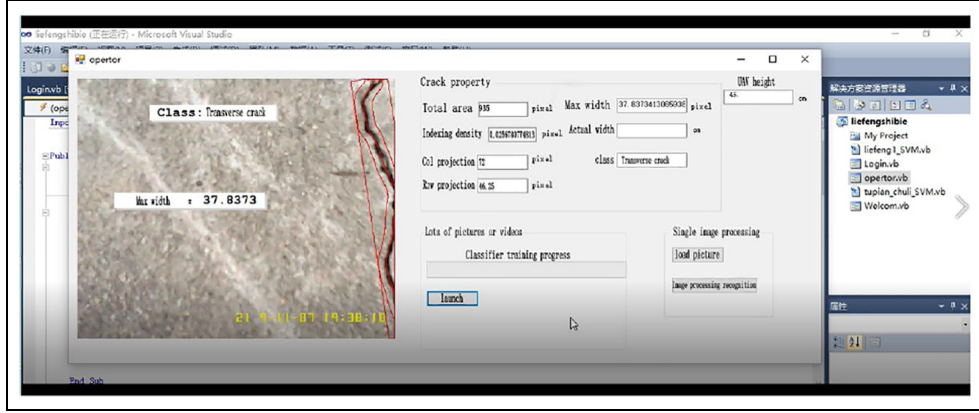
Table 2. The specifications of low-cost UAV.

Component name	Components type	Quantity
Microcontroller	STM32F407VGT6	1
Barometer	MEAS MS5611	1
Accelerometer	MPU6050	1
Magnetometer	HMC5883L	1
Wi-Fi module	ESP8266	1
Flash	M24128-BWDW6TP	1
Ultrasonic module	HC-SR04	1
Storage module	Micro SD card	1
Screen	0.96 inch OLED screen	1
LED	RGB full color LED	1
ESC brushless motor	HTI-D20-491101	4
Brushless motor	A2212-13 980 kV	4
GPS module	Radiolink M8N GPS SE100	1
Figure transmission module	VT58 5.8G 40CH FPV	1
Power supply module	3S lithium battery (5200 mAh 11.1 V)	1

UAV: unmanned aerial vehicle; LED: light-emitting diode; OLED: organic light-emitting diode; ESC: electronic spark control; GPS: global positioning system.

module is used to handle the debugging of the flight control algorithm. Power supply module is used to supply the entire system with power. Figure 3 shows the hardware and real environment of the low-cost UAV. The detailed specifications of components are listed in Table 2. Furthermore, to illustrate the detailed specification of the camera in our designed UAV, we listed all of the parameters of the camera which costs around US\$80 in Table 3.

With the installed Raspberry Pi 3 Model B in our developed UAV, the UAV only needs to be powered by the external direct current power supply of 5 V and 2 A, and the power supply can be achieved by ordinary batteries. The camera is connected to the Raspberry Pi.



**Figure 4.** System user interface in computer.

**Table 3.** The specifications of low-cost camera.

Components name	Components parameter
Product name	1000TVL camera lens
Input voltage	15 V
Power consumption	55 mA
Lux	0.08 Lux/F1.2 1/3" CCD sensor
Format	NTSC/PAL switch (free)
Resolution	1000TVL ultra high definition
Lens	The field of view is 110°, with 2.8 mm infrared coating
Operating temperature	0°C~40°C
Size	28 × 24.5 × 17.5 mm camera

CCD: charge-coupled device.

Meanwhile, a Linux system is installed on the Raspberry Pi, and our CCPM pre-processed algorithm is implemented in this system. Through the Wi-Fi network, the Raspberry Pi is connected with the computer, which has Intel® Core™ i5-8250U CPU @1.60 Hz, 1.80 GHz and 4.0 GB RAM, for the real-time image classification via the trained SVM model in computer side. Furthermore, the remote desktop software on the computer can control the Raspberry Pi to detect cracks in real time. Figure 4 shows that the system user interface runs in the computer side.

The computer controls the Raspberry Pi to collect crack images, and it also controls the camera's image capture time. In our case, the camera's capture rate is 10 images per second. Moreover, the capturing speed of the crack images and the appropriate image data options can be configured autonomously. In real-time detection, the speed of one shot can reach around 1 s.

### New algorithm design for the proposed method

According to Algorithm 1, we implement the new method using C language. In our new method, we use the crack

### Algorithm 1: Vision-based crack inspection algorithm

**Input:** Original Color Image Matrix,  $I = (H_o \times W_o \times 3)$

**Output:** Accuracy A

**Step 1:** Convert Color Image to Binary Image.

GrayPixel  $x_{i,j}$  = ConvertImage.OrigToGray ( $I(H_o \times W_o \times 3)$ );

BinaryPixel  $x_{i,j}$  = CovertImage.GrayToBin (GrayPixel  $x_{i,j}$ );

**Step 2:** Detect the contour in grayscale binary image.

Compute smallest eigenvalue  $\alpha_{min}$  with equation (2);

Compute minimum grayscale value  $L(p_i)$  with equation (2);

**Step 3:** Crack Feature Extraction.

Compute  $\phi$  with equations (3) and (4);

Compute  $H$  with equation (6);

**Step 4:** Cracks classified by the SVM algorithm (SMO).

Training set  $\bar{x}_i$ ;

Label set  $\bar{y}_i = H$ ;

**Initialize:**  $\alpha_i = 0, f_i = -b$ ;

**Compute:**  $L, H$  using equations (11) and (12);

**Update:**  $\alpha_1, \alpha_2$  using equations (13) and (14);

**Repeat:**

Update  $f_1, f_2$  using equation (15);

Compute  $L_1, H_1$  using equation (15);

Update  $\alpha_1, \alpha_2$  using equations (13) and (14);

**Until:**  $f_1$  converge to  $f_2$ ;

**Update:**  $b_1, b_2$  using equations (16) and (17).

Store new  $\alpha_1$  and  $\alpha_2$ .

**Step 5:** Recognize the cracks.

Input the new samples  $S$ ;

Call the SVM algorithm to identify the input sample;

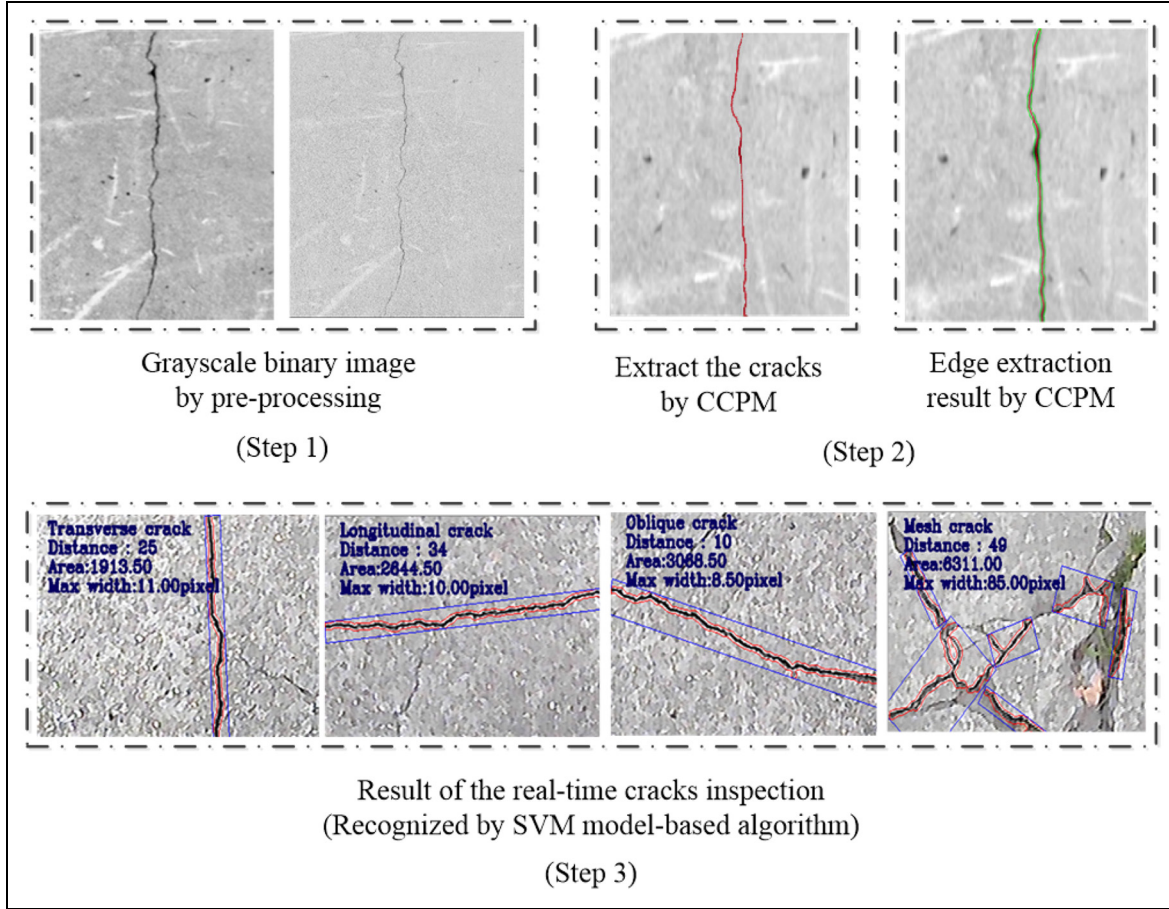
Calculate the inspection accuracy;

Output the crack inspection accuracy A;

projection characteristics in X- and Y-coordinates, crack area, and distributed density as the input features of SVM model and crack type as the output label for SVM model. The correct and classified label of each sample is compared to compute the accuracy of the inspection.

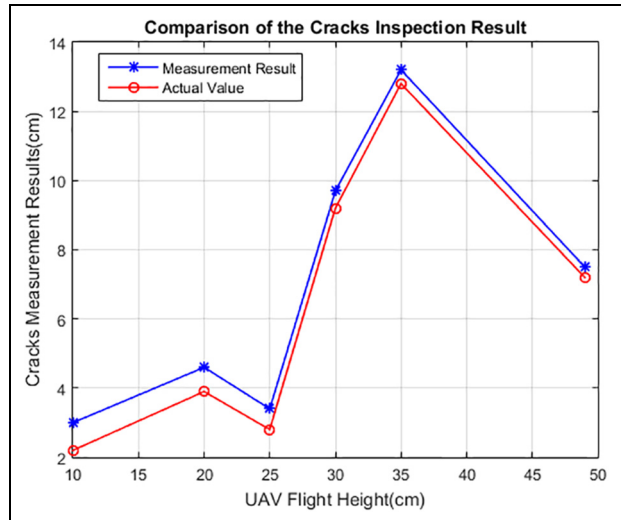
### Result discussion

In our experiment, we inspected four types of cracks—transverse, longitudinal, oblique, and mesh—in



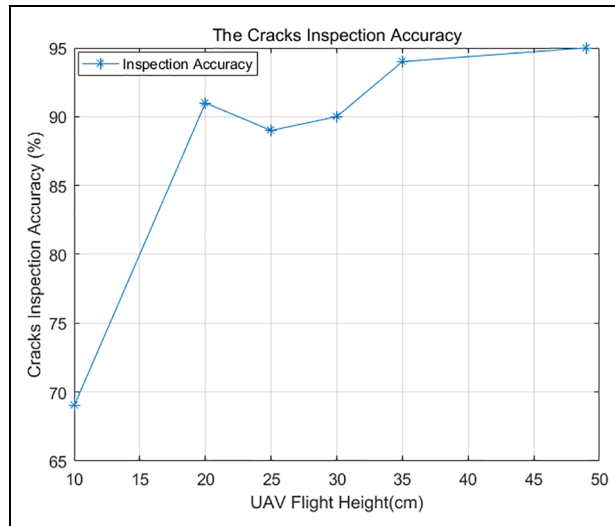
**Figure 5.** The whole experimental process.

different flight heights of UAVs. These inspection images are captured by the UAV in real time and translated to the terminal devices—such as laptops, pads, smart phones, and others—via Wi-Fi network. Figure 5 shows the example images and result of the real-time crack inspection by the proposed method. The experimental result in Figure 6 shows that the measurement size of each crack is quite close to the actual size. Moreover, Figure 7 demonstrates that the accuracy of the inspection of different cracks generally improves with the flight height increase. The reason lies in the different optical (lens) distortions, which are generated by the optical design of lenses in the different heights of UAVs. Moreover, Figures 5 and 6 show small part of the test results, which are the results of the detection after training, from the experiment. Table 4 presents the number of training data, test data, and recognition accuracy, which illustrates more details. Moreover, in the model training process, we randomly and independently divide all image data into two data sets with the ratio of 7:3 (70% image data for training and 30% image data for testing). Meanwhile, we also use the  $k$ -fold ( $k = 10$ ) cross-validation approach to select the



**Figure 6.** Results of crack inspection experiment.

detection model with highest performance and avoid overfitting. The more detailed data set and testing results are listed in Table 4.



**Figure 7.** Crack inspection accuracy.

**Table 4.** The detail of training data and classification accuracy.

	Transverse crack	Vertical crack	Oblique crack	Mesh crack
Training data	280	280	280	280
Test data	120	120	120	120
The correct quantity of classification	98	107	106	103
Classification correct rate	81.67%	89.17%	88.33%	85.83%

Overall, the result illustrates the accuracy, applicability, and robustness of the proposed methods in the different flight heights of the UAV. The only minor issue is that SVM has slightly lower detection accuracy for mesh cracks than for other crack types. The reason may be that mesh cracks are more sensitive to the environment noise and have less crack length. Consequently, some features of mesh cracks may be easier to be filtered out in the image pre-processing. So, mesh cracks have more probability to be undetected or misclassified.

## Conclusion and future work

Till now, a few noncontact and vision-based SHM methods have been effectively utilized and integrated with low-cost UAVs to perform the real-time crack inspection. To solve this challenge, a novel low-cost UAV-enabled, vision and SVM-based crack inspection method has been designed and implemented to inspect the cracks of concrete structure in real time. The

working procedures include the three steps: pre-processing of the image acquired by the UAV, extraction of crack characters from the pre-processed images using CCPM, and the crack classification via SVM. The results from the bridge concrete inspection experiment further validate the effectiveness and robustness of this proposed method. Therefore, this article provides an exemplary research about the state-of-the-art computer vision, machine learning technology, and low-cost UAV application in noncontact SHM. The proposed method realizes the real-time crack inspection and process automation, and displays the significant advantages over traditional noncontact SHM methods.


Future research includes the four following aspects:

1. The detailed analysis of computational time and space complexity of the proposed vision-based method to implement seamless integration with low-cost UAVs and reduce system failures.
2. Currently, UAVs are required to have the images taken perpendicular to the concrete surface. Utilizing and optimizing our developed algorithm to detect cracks from images taken by other angles or directions need further study in the future.
3. The solutions to improve CCPM, so that its robustness can be further enhanced especially for the low-quality images captured with severe noise, dirty spots, rusts, shades, UAV jitter, or in extremely poor environmental conditions.
4. The crack classification using different unsupervised and supervised learning algorithms—such as genetic algorithm, boosting, random forest, neural network,<sup>50</sup> and deep learning—to simulate human vision and logical thinking,<sup>37</sup> and to improve the accuracy of the mesh crack detection.

## Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

## ORCID iD

Gangbing Song  <https://orcid.org/0000-0001-5135-5555>

## References

1. Harms T, Sedigh S and Bastianini F. Structural health monitoring of bridges using wireless sensor networks. *IEEE Instrum Meas Mag* 2010; 6(13): 14–18.
2. Tessari G, Riccardi P and Pasquali P. *Monitoring the ongoing deformation and seasonal behaviour affecting Mosul Dam through space-borne SAR data*. Washington, DC: American Geophysical Union, 2017.



3. Peng J, Hu S, Zhang J, et al. Influence of cracks on chloride diffusivity in concrete: a five-phase mesoscale model approach. *Construct Build Mater* 2019; 197: 587–596.
4. Fan W and Qiao P. Vibration-based damage identification methods: a review and comparative study. *Struct Health Monit* 2011; 10(1): 83–111.
5. Ou J and Li H. Structural health monitoring in mainland China: review and future trends. *Struct Health Monit* 2010; 9(3): 219–231.
6. Qiao P and Cao M. Waveform fractal dimension for mode shape-based damage identification of beam-type structures. *Int J Solid Struct* 2008; 45(22–23): 5946–5961.
7. Wang J and Qiao P. On irregularity-based damage detection method for cracked beams. *Int J Solid Struct* 2008; 45(2): 688–704.
8. Baptista FG and Vieira Filho J. A new impedance measurement system for PZT-based structural health monitoring. *IEEE T Instrum Meas* 2009; 58(10): 3602–3608.
9. Yu Y, Ou J, Zhang J, et al. Development of wireless MEMS inclination sensor system for swing monitoring of large-scale hook structures. *IEEE T Ind Electr* 2009; 56(4): 1072–1078.
10. Lu G, Li Y, Zhou M, et al. Detecting damage size and shape in a plate structure using PZT transducer array. *J Aeronaut Eng* 2018; 31(5): 04018075.
11. Liao WI, Wang JX, Song G, et al. Structural health monitoring of concrete columns subjected to seismic excitations using piezoceramic-based sensors. *Smart Mater Struct* 2011; 20(12): 125015.
12. Kong Q, Robert RH, Silva P, et al. Cyclic crack monitoring of a reinforced concrete column under simulated pseudo-dynamic loading using piezoceramic-based smart aggregates. *Appl Sci* 2016; 6(11): 341.
13. Qiao P, Lestari W, Shah MG, et al. Dynamics-based damage detection of composite laminated beams using contact and noncontact measurement systems. *J Compos Mater* 2007; 41(10): 1217–1252.
14. Li P, Gu H, Song G, et al. Concrete structural health monitoring using piezoceramic-based wireless sensor networks. *Smart Struct Syst* 2010; 6(56): 731–748.
15. Luo M, Li W, Hei C, et al. Concrete infill monitoring in concrete-filled FRP tubes using a PZT-based ultrasonic time-of-flight method. *Sensors* 2016; 16(12): 2083.
16. Magno M, Boyle D, Brunelli D, et al. Extended wireless monitoring through intelligent hybrid energy supply. *IEEE T Ind Electron* 2014; 61(4): 1871–1881.
17. Davoudi R, Miller GR and Kutz JN. Computer vision based inspection approach to predict damage state and load level for RC members. *Struct Health Monit* 2017, <http://www.dpi-proceedings.com/index.php/shm2017/article/view/14225>
18. Huang JZ, Li DS, Li HN, et al. Damage identification of a large cable-stayed bridge with novel cointegrated Kalman filter method under changing environments. *Struct Control Health Monit* 2018; 25(5): e2152.
19. Reagan D. *Unmanned aerial vehicle measurement using three dimensional digital image correlation to perform bridge structural health monitoring*. Doctoral Dissertation, University of Massachusetts Lowell, Lowell, MA, 2017.
20. Li H, Ou J, Zhang X, et al. Research and practice of health monitoring for long-span bridges in the mainland of China. *Smart Struct Syst* 2015; 15(3): 555–576.
21. Khaloo A, Lattanzi D, Jachimowicz A, et al. Utilizing UAV and 3D computer vision for visual inspection of a large gravity dam. *Frontier Built Environ* 2018; 4: 31.
22. Catbas FN and Aktan AE. Condition and damage assessment: issues and some promising indices. *J Struct Eng* 2002; 128(8): 1026–1036.
23. Lapoutre G. Acoustic emission monitoring: of offshore wind turbine support structures for detection and localization of fatigue crack growth, 2017, <https://repository.tudelft.nl/islandora/object/uuid%3Aa5d405b1-92a7-4320-af33-da94602d87ed>
24. Hallermann N, Morgenthal G and Rodehorst V. Vision-based deformation monitoring of large scale structures using unmanned aerial systems. In: *Proceedings of the IABSE symposium report international association for bridge and structural engineering*, Vol. 102, Madrid, 3–5 September 2014, pp. 2852–2859. International Association for Bridge and Structural Engineering.
25. Kim H, Lee J, Ahn E, et al. Concrete crack identification using a UAV incorporating hybrid image processing. *Sensors* 2017; 17(9): 2052.
26. Hoskere V, Narazaki Y, Hoang T, et al. Vision-based structural inspection using multiscale deep convolutional neural networks, 2018, <https://arxiv.org/ftp/arxiv/papers/1805/1805.01055.pdf>
27. Lattanzi D and Miller G. Review of robotic infrastructure inspection systems. *J Infrastruct Syst* 2017; 23(3): 04017004.
28. Khaloo A, Lattanzi D, Cunningham K, et al. Unmanned aerial vehicle inspection of the placer river trail bridge through image-based 3D modelling. *Struct Infrastruct Eng* 2017; 14: 124–136.
29. Brooks C, Dobson RJ, Banach DM, et al. *Evaluating the use of unmanned aerial vehicles for transportation purposes* (No. RC-1616). Ann Arbor, MI: Michigan Tech Research Institute, 2015.
30. Hernandez I, Fields T and Kevern J. Overcoming the challenges of using unmanned aircraft for bridge inspections. In: *Proceedings of the AIAA atmospheric flight mechanics conference*, San Diego, 4–8 January 2016, pp. 33–96. Reston, VA: AIAA.
31. Pereira FC and Pereira CE. Embedded image processing systems for automatic recognition of cracks using UAVs. *IFAC-PapersOnline* 2015; 48(10): 16–21.
32. Ellenberg A, Branco L, Krick A, et al. Use of unmanned aerial vehicle for quantitative infrastructure evaluation. *J Infrastruct Syst* 2014; 21(3): 04014054.
33. Metni N and Hamel T. A UAV for bridge inspection: visual servoing control law with orientation limits. *Autom Construct* 2007; 17(1): 3–10.
34. Sankarasrinivasan S, Balasubramanian E, Karthik K, et al. Health monitoring of civil structures with integrated UAV and image processing system. *Proced Comput Sci* 2015(54): 508–515.

35. Choi SS and Kim EK. Building crack inspection using small UAV. In: *Proceedings of the 17th international conference on advanced communication technology (ICACT)*, Seoul, 1–3 July 2015, pp. 235–238. New York: IEEE.
36. Forsyth D and Ponce J. *Computer vision: a modern approach*, vol. 6. London: Pearson, 2011.
37. Bao Y, Tang Z, Li H, et al. Computer vision and deep learning-based data anomaly detection method for structural health monitoring. *Struct Health Monit* 2019; 18(2): 401–421.
38. Wang L and Zhang Z. Automatic detection of wind turbine blade surface cracks based on UAV-taken images. *IEEE T Ind Electron* 2017; 64(9): 7293–7303.
39. Eschmann C, Kuo CM, Kuo CH, et al. High-resolution multisensor infrastructure inspection with unmanned aircraft systems. *Int Arch Photogram Remote Sens Spatial Inf Sci* 2013(2): 125–129.
40. Kang D and Cha YJ. Autonomous UAVs for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging. *Comput-Aided Civ Infrastruct Eng* 2018; 33: 885–902.
41. Lei B, Wang N, Xu P, et al. New crack detection method for bridge inspection using UAV incorporating image processing. *J Aerosp Eng* 2018; 31(5): 04018058.
42. Suzuki S. Topological structural analysis of digitized binary images by border following. *Comput Vis Graph Image Pr* 1985; 30(1): 32–46.
43. Yang JN and Lin S. Identification of parametric variations of structures based on least squares estimation and adaptive tracking technique. *J Eng Mech* 2005; 131(3): 290–298.
44. GB 50144-2008. Standard for appraisal of reliability of industrial buildings and structures.
45. Adankon MM and Cheriet M. Support vector machine. In: Li S and Jain A (eds) *Encyclopedia of biometrics*. New York: Springer, 2009, pp. 1303–1308.
46. Berwick R. An idiot's guide to support vector machines (SVMs), 2003, <http://web.mit.edu/6.034/wwwbob/svm.pdf>
47. Zeng ZQ, Yu HB, Xu HR, et al. Fast training support vector machines using parallel sequential minimal optimization. In: *Proceedings of the 3rd international conference on intelligent system and knowledge engineering*, Vol. 1, Xiamen, China, 17–19 November 2008, pp. 997–1001. New York: IEEE.
48. Izmailov AF and Solodov MV. Karush-Kuhn-Tucker systems: regularity conditions, error bounds and a class of Newton-type methods. *Math Program* 2003; 95(3): 631–650.
49. Trafalis TB and Ince H. Support vector machine for regression and applications to financial forecasting. In: *Proceedings of the IEEE-INNS-ENNS international joint conference on neural networks*, Vol. 6, Como, 27 July 2000, pp. 348–353. New York: IEEE.
50. Li H, Bao Y and Ou J. Structural damage identification based on integration of information fusion and Shannon entropy. *Mech Syst Signal Pr* 2008; 22(6): 1427–1440.