Automatic Crack Detection in Built Infrastructure Using Unmanned Aerial Vehicles

M.D. Phung, T.H. Dinh, V.T. Hoang and Q.P. Ha

Faculty of Engineering and Information Technology, University of Technology Sydney, Australia
E-mail: Manhduong.Phung@uts.edu.au, Quang.Ha@uts.edu.au

Abstract -

This paper addresses the problem of crack detection which is essential for health monitoring of built infrastructure. Our approach includes two stages, data collection using unmanned aerial vehicles (UAVs) and crack detection using histogram analysis. For the data collection, a 3D model of the structure is first created by using laser scanners. Based on the model, geometric properties are extracted to generate way points necessary for navigating the UAV to take images of the structure. Then, our next step is to stick together those obtained images from the overlapped field of view. The resulting image is then clustered by histogram analysis and peak detection. Potential cracks are finally identified by using locally adaptive thresholds. The whole process is automatically carried out so that the inspection time is significantly improved while safety hazards can be minimised. A prototypical system has been developed for evaluation and experimental results are included.

Keywords -

Crack detection; Unmanned aerial vehicles; Health monitoring; Image stitching

1 Introduction

Crack detection is essential in health monitoring of infrastructure. Road sections containing a high density of cracks at the surface should be periodically maintained to ensure the safe operation of vehicles. Crack type, size and the level of severity need to be identified for this task. In concrete bridges, cracking caused by natural processes is inevitable and may result in malfunctioning of the entire bridge, or even collapse. It opens access for water, deicing salts and other corrosive chemicals to penetrate through the bridge deck and over time causes damages to internal bridge structures. Early identification of cracks is thus vitally important to maintain the service life of bridges.

To automatically detect cracks, both colour and geometry information need to be acquired with sufficient quality using unmanned ground vehicles [10, 20] or a group of mobile robots [26]. However, complexity and roughness remain challenges for surface inspection using this approach. Recently, unmanned aerial vehicles (UAVs) have been developed as an alternative owing to their flexibility in the operation space and ability to carry specialised sensory equipment. In [6], a micro air vehicle system was employed to scan buildings using a high resolution camera. Images taken within a restricted location were then stitched with sufficient quality for crack and damage detection. An advanced UAV system was developed to evaluate the state of historical monuments [8] whose captured images revealed after processing some damages in several monuments. In [14], a control system for navigating the UAV in unknown 3D environments was used to monitor and maintain bridges. UAVs were also used to inspect and monitor oil-gas pipelines, roads, power generation grids and other essential infrastructure [21].

For surface detection, computer-vision based techniques are widely used to detect cracks due to its robustness and cost efficiency [11]. These techniques in general can be categorised into the wavelet transform, minimal path selection, machine learning, edge detection and intensity thresholding. For example, a separable 2D continuous wavelet transform is employed in [25], using complex coefficient maps for crack segmentation. An improvement of the wavelet-based pavement distress detection can be achieved by combining the Wavelet-Radon transform and dynamic neural network thresholding [15]. These techniques however do not consider the geometric characteristics (orientation, continuity and connectedness) of the cracks and may wrongly detect the candidates with low continuity or high curvature.

Owing to the ability to effectively identify high-level geometric information, the minimal path principle can be applied in surface crack detection. In [16], the free-form anisotropy is able to handle almost all crack characteristics in a segmentation step. An improvement of the minimal path technique is presented in [9], having the capability to detect cracks without prior knowledge of endpoints. A fully unsupervised approach is proposed in [3], where a refined artefact filtering step is introduced to estimate the width of the crack. However, the main drawback of the minimal path approach is high computation time involved.

With the explosive development of image data, machine learning-based methods [17, 23] have been effectively used for surface crack detection. In [17], a multi-level pattern recognition system is developed to address

image blocks containing cracks and then evaluate their geometry such as length and width. Random structured forests are used in [13] and [23] to extract information for crack descriptors from the image background. Nevertheless, these methods are greatly dependent on the training data, which is labour-intensive with manual labelling in the training and validation steps.

For determining potential cracks from bridge decks, edge detection techniques is commonly used [2, 12]. Four edge-detection algorithms are surveyed in [1], where the Haar Wavelet method is identified as most reliable, compared to the gradient-based (Sobel and Canny) and frequency-based (fast Fourier Transformation) algorithms. These methods, however, only perform well under uniform illumination and low noise conditions.

In crack regions that are consistently darker than their surrounding areas, intensity-thresholding methods [18] are more appropriate due to their compactness. A Bayesian classification technique together with a morphological opening operation and thresholding is applied to segment and classify defects from underground concrete pipes into various classes such as cracks, holes, laterals and joints [24]. A two-step method is proposed in [7] where a locally adaptive thresholding is used together with a median filter and a multi-scale line filter to emphasise the line structure and detect crack candidates. As thresholding is noise sensitive, it is often used with other techniques such as morphological or linear filtering to improve robustness.

In this paper, we propose a system to inspect built infrastructure for automatic crack detection. A 3D model of the object is first created and its geometric features are extracted to generate a path for UAV navigation. While following the planned path, the UAV takes images of the suspected surfaces. Those images are then stitched and processed based on histogram analysis. For this task, we develop a peak detection algorithm for image clustering and a locally adjustable thresholding technique for crack detection. A number of experiments have been carried out and the detection results are promising to apply in real time applications.

2 Data Collection Using UAV

The goal of data collection is to acquire adequate geometric and surface information of the object to be used in post processing for detecting potential cracks, and also for navigation of the UAV itself [19].

2.1 Point cloud 3D modelling

In this step, laser scanners are used to acquire range information from different positions of the structure to be inspected and represent them as point clouds. Those point clouds are then merged one by one in a process called registration to create a 3D model. In the registration, overlapping points corresponding to the same part of the structure appearing among the point clouds are first identified. Let P_a and P_b be the point clouds recorded at positions $\boldsymbol{a}$ and $\boldsymbol{b}$. The overlapping points are determined by:

$$\|(\boldsymbol{x}_i^a - \boldsymbol{a}) - (\boldsymbol{x}_j^b - b)\| < \tau, \tag{1}$$

where $\boldsymbol{x}_i^a \in P_a$ and $\boldsymbol{x}_i^b \in P_b$ are overlapping points, considered in a close neighbourhood, and τ is a pre-defined distance threshold. The alignment of point clouds is then obtained by applying an iterative closest point algorithm to find a transformation that minimises the distances between them.

2.2 Geometric feature extraction

Planar surfaces are often the main target to be inspected so they need to be extracted from the point cloud. Given a plane's equation ($ax + by + cz + d = 0$), then $M = [a, b, c, d]^T$ is the parameter vector to be identified. For this, a random sample consensus (RANSAC) algorithm is applied with some augmentations, including a noise filter to remove sparse outliers, voxelisation to equalise the point density, and clustering to trim out the isolated groups before applying RANSAC.

After detecting the surfaces, their boundaries are determined by using a convex hull algorithm. The remaining point cloud P is then clustered into small groups as obstacles to be considered in path planning. Here, for a given positive constant $\epsilon > 0$, a cluster is defined as a set of points:

$$C = \{p_i \in P \mid \min\|p_i - p_j\| > \epsilon\}, \tag{2}$$

for any other point not belonging to the cluster, i.e., $p_j \notin C$.

2.3 Path planning for colour image acquisition

Given surfaces to be inspected, a list of waypoints needs to be created to navigate the UAV. There are two types of waypoints, one corresponds to shooting points for taking images and the others serve as intermediate points for path following and avoiding obstacles. They are generated by first splitting the operating environment into voxels. A status of free or occupation is then defined for each voxel to indicate the existence or not of obstacles in that voxel. The shooting points are then computed based on intrinsic parameters such as camera focal length, surface area and minimum resolution. An A-star algorithm is finally applied to find the shortest path between shooting points. In each step, the cost to move from one voxel to another in the neighbourhood is computed as:

$$C(k, l, m) = a_1 k^2 + a_2 l^2 + a_3 m^2, \tag{3}$$

where coordinates $k, l, m \in \{-1, 0, 1\}$ indicate the neighbour position, and the coefficients a_1, a_2 and a_3 assign a particular weight to each direction.

The generated waypoints are used as references for motion control of the UAV. Typically, the controller are built in the flight operating system so that the control task can be simplified. To collect images of sufficient quality, a gimbal is used to eliminate vibration and adds more degree of freedoms to the system to shoot images perpendicularly to the inspected surface.
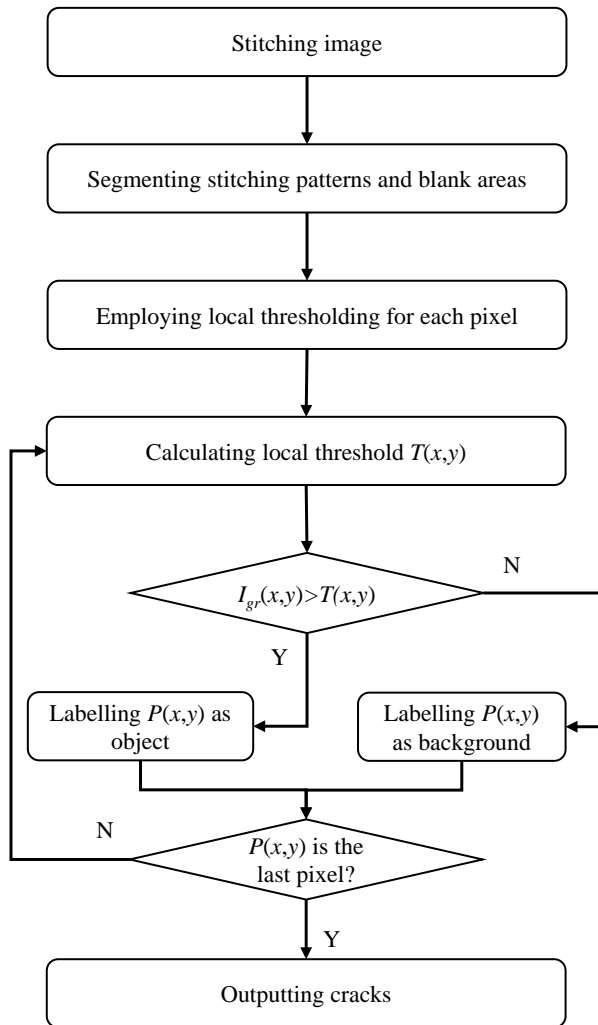
Figure 1: Crack detection flowchart.

3 Thresholding-Based Crack Detection

A crack detection algorithm is developed to process the collected data based on analysing their histogram information. Its flowchart including image stitching, pattern removal and crack detection steps is shown in Fig. 1.

3.1 Image stitching

Each image taken by the UAV only covers a small area of the inspected surface. Stitching them to create a panoramic image is thus essential for crack detection. This stage requires a certain level of overlapping between consecutive images and corresponding features. An invariant feature based approach [4] is employed, consisting of scale-invariant feature transform (SIFT) feature extraction, homography computation and image matching verification. If the surface to be inspected is homogeneous, manual patterns can be added to enhance the stitching performance and removed afterwards.

3.2 Stitching pattern removal

Before processing the stitched image, it is important to remove the patterns used in stitching to enhance the detection accuracy. A stitched image typically consists of three elements, namely crack area, stitching patterns and blank area caused by alignment drifts. Since their luminance varies in accordance with the sunlight condition, our peak detection algorithm [5] is first employed to detect dominant peaks corresponding to these elements. The thresholds t_1 and t_2 to segment the surfaces to be inspected are calculated as the intensity average:

$$\begin{cases} t_1 = \dfrac{i_b + i_w}{2} \\ t_2 = \dfrac{i_w + i_p}{2}, \end{cases} \tag{4}$$

where i_b, i_w and i_p are the intensity value of peaks corresponding to the blank areas, the surface and stitching patterns, respectively. The stitching patterns are then identified based on the histogram as follows:

$$I_{gr}(x, y) = \beta \quad \text{if} \quad \begin{bmatrix} I_r(x, y) < t_1 \\ I_r(x, y) > t_2, \end{bmatrix} \tag{5}$$

where $I_{gr}(x, y)$ and $I_r(x, y)$ are respectively the grey intensity and the red channel intensity of the stitched image at point $P(x, y)$, and β is the intensity value chosen to distinguish the pattern with other parts of the image. As the intensity at crack structures is typically smaller than at non-defect areas of inspected surfaces, here β is set to 255 for adequately filtering out those features without information loss.

3.3 Crack detection

In an outdoor environment subject to varying lighting conditions, the global thresholding method [5] is extended to be able to extract all crack candidates with large variation of intensities. The approach proposed here, is first to take the advantage of the automatic peak detection for pre-processing the image to retain only the background

Figure 2: The 3DR Solo drone with remote controller and ground control station.

Figure 3: Collecting data of the bridge.

Figure 4: Flight path to collecting data of the bridge.

and line-like objects, and then to apply locally-adjusted thresholds [22] to identify potential cracks. Differing from the global approach, here the threshold is computed for each pixel based on the grey intensities of its neighbours. Let us consider (x_i, y_i), $i = 1, 2, ..., N^2$, in the neighbourhood of pixel $P(x, y)$ determined by using an $N \times N$- window and $m(x, y)$ and $s(x, y)$ respectively as the mean and standard deviation in that window:

$$m(x, y) = \frac{1}{N^2} \sum_{i=1}^{N^2} I_{gr}(x_i, y_i), \qquad (6)$$

$$s(x, y) = \sqrt{\frac{1}{N^2} \sum_{i=1}^{N^2} (m(x, y) - I_{gr}(x_i, y_i))^2}. \qquad (7)$$

The threshold for pixel $P(x, y)$ is then computed as:

$$T(x, y) = m(x, y) \left[1 + k \left(\frac{s(x, y)}{R} - 1 \right) \right], \qquad (8)$$

where R is the dynamic range of standard deviations and k is a tunable parameter used to adjust the influence of standard deviation. Each pixel $P(x, y)$ is then evaluated against its threshold $T(x, y)$. A pixel $P(x, y)$ is considered as belonging to a crack if its grey intensity $I_{gr}(x, y)$ is higher than the computed local threshold, or lying in the background otherwise.

3.4 Experiments

Experiments have been conducted to evaluate the proposed approach. The UAV used is a quadcopter, the 3DR Solo drone shown in Fig.2. It is equipped with three processors, two are Cortex M4 168 MHz running Pixhawk firmware for low-level control and the other is an ARM Cortex A9 running Arducopter flight operating system. The camera used is Hero 4 with the focal length of 34.4 mm and resolution of 12 megapixels, attached to a three-axis gimbal with one degree-of-freedom for controlling
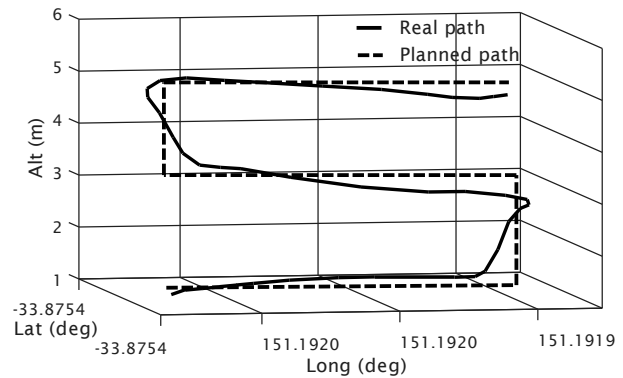
its yaw angle. The programming is carried out through the ground control station called Mission Planner and uploaded to the UAV.

3.4.1 Data collection results

Figures 3 and 4 show a practical path used to collect data of a bridge. There are mismatches between the planned and real paths which are inevitable due to localisation errors caused the built-in global positioning system. Nevertheless, in our experiments, the number of satellites detected ranging from 9 to 11 causing the error less than 1.5 m which is relatively small. For data collection, this error can be compensated by reducing the UAV's speed while increasing the number of shooting points. It is also noted that although the planned path shown in Fig. 4 is rather ideal and does not consider the dynamic constraints of the UAV, which is beyond the scope of this study, the path tracking error remains however in an acceptable tolerance for the inspection purpose.

3.4.2 Crack detection results

To evaluate the crack detection algorithm, an wooden wall was set up from nine panels, each having the size of 600 mm × 900 mm × 3 mm. Those panels were hanged on a frame and joined together using twisted ties as shown
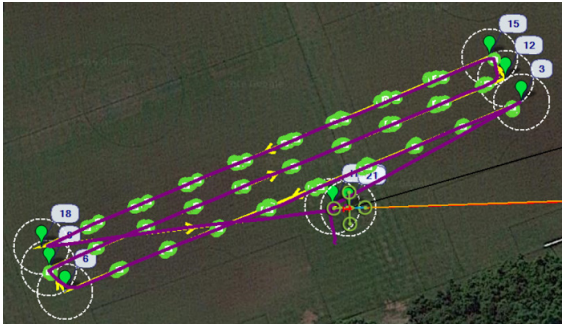
Figure 5: Flight path used to inspect the artificial wall.

in Fig. 6a. Eighteen patterns were manually stuck on top of the panels for image stitching. The pink colour was chosen for the patterns due to its large difference from the colour of wooden panels. Two cracks were deliberately formed by a manual impact applied on two panels, one at the middle and one at the bottom right. The UAV was programmed to follow pre-defined waypoints to take images of the surface, as shown in Fig. 5. The values of R and k in (8) are chosen to be 128 and 0.5, respectively.

Figure 6b shows the stitched image. It can be seen that there are almost no distortions compared to the original wall. However, missing pixels caused by alignment drifts still appear. This issue can be resolved by employing gain compensation and multi-band blending techniques. Figure 6c shows the result of filtered stitching patterns. They are all well isolated from the stitched image demonstrating the effectiveness of the proposed automatic peak detection algorithm.

Figure 7a presents the crack detection results using the global thresholding method [5]. It can be seen that there are two crack candidates appearing along with horizontal and vertical lines of the tiled wooden panels. However, the middle crack is not well detected due to the variation in the grey intensity. If the threshold is increased to better capture the crack image, then parts of the wall that are exposed to the sunlight are not adequately filtered as shown in Fig. 7b. This problem can be solved using the locally adjusted thresholding method as shown in Fig. 7c.

4   Conclusion

In this work, we have developed an automatic crack detection system for infrastructure monitoring. By using UAVs, the system is capable of inspection of poorly accessible structures such as dams, culverts or bridges. A number of sensors have been integrated into the system allowing it to acquire geometric and colour information of the inspected surfaces. For data processing, we have developed computer vision based algorithms to create 3D models of the structure, extract features, plan navigation paths, stitch images and detect potential cracks. A number of experiments have been conducted with all cracks de-

tected in real time. Future work will focus on further improvements of the recognition algorithms to better identify crack properties such as the length, width and orientation.

Ackowledgement

References

[1] I. Abdel-Qader, O. Abudayyeh, and M. Kelly. Analysis of edge-detection techniques for crack identification in bridges. Journal of Computing in Civil Engineering, 17(4):255–263, 2003.

[2] R. S. Adhikari, O. Moselhi, and A. Bagchi. Image-based retrieval of concrete crack properties for bridge inspection. Automation in Construction, 39:180 – 194, 2014.

[3] R. Amhaz, S. Chambon, J. Idier, and V. Baltazart. Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection. IEEE Transactions on Intelligent Transportation Systems, 17(10):2718–2729, Oct 2016.

[4] M. Brown and D. Lowe. Automatic panoramic image stitching using invariant features. International journal of computer vision, 74(1):59–73, 2007.

[5] T. H. Dinh, Q. P. Ha, and H. M. La. Computer vision-based method for concrete crack detection. In Proceedings of the 14th International Conference on Control, Automation, Robotics and Vision (ICARCV 2016), pages 1–6, Nov 2016.

[6] C. Eschmann, C.-H Kuo, C.-M Kuo, and C. Boller. Unmanned aircraft systems for remote building inspection and monitoring. In Proceedings of the 6th European Workshop on Structural Health Monitoring, pages 1–8, 2012.

[7] Y. Fujita and Y. Hamamoto. A robust automatic crack detection method from noisy concrete surfaces. Machine Vision and Applications, 22(2):245–254, 2011.

[8] N. Hallermann and G. Morgenthal. Un-manned aerial vehicles (uav) for the assessment of existing structures. In Proceedings of the 36th IABSE Symposium, 101(14):266–267, 2013.
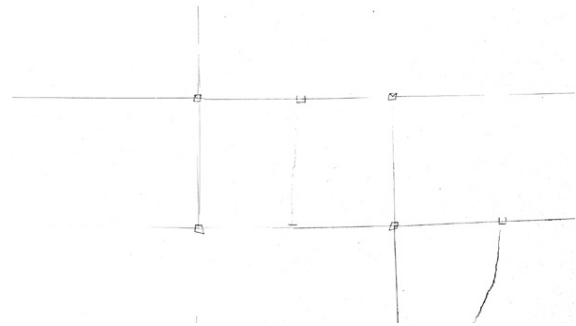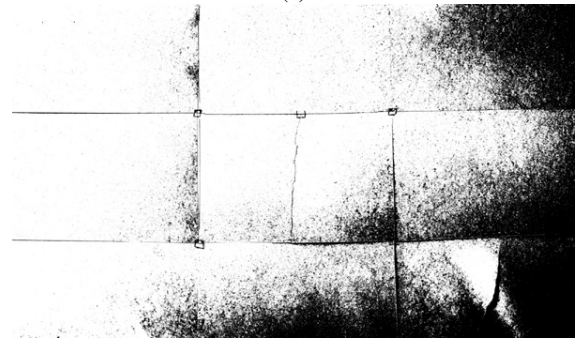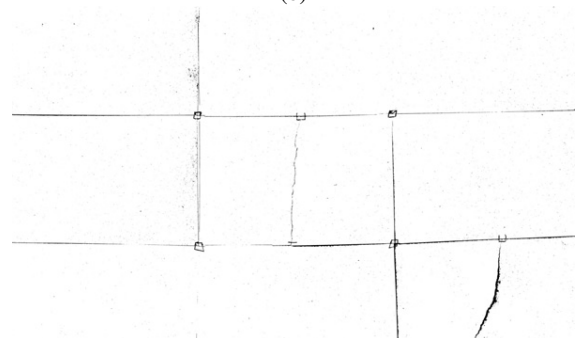
(a)

(b)
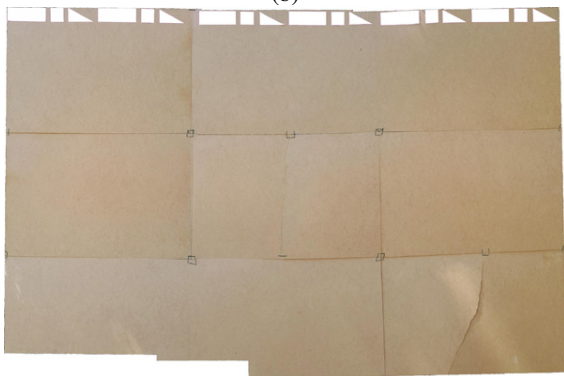
(c)

Figure 6: Image stitching and segmentation:
(a) UAV taking photos of the artificial wall (b) Stitched image, (c) Segmented patterns and blank areas.

(a)

(b)

(c)

Figure 7: Crack detection result: (a) Global thresholding at T = 125, (b) Global thresholding at T = 155, (c) Locally apaptive thresholding.

[9] V. Kaul, A. Yezzi, and Y. Tsai. Detecting curves with unknown endpoints and arbitrary topology using minimal paths. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(10):1952–1965, Oct 2012.

[10] S. Kim, M. Spenko, S. Trujillo, B. Heyneman, D. Santos, and M. R. Cutkosky. Smooth vertical surface climbing with directional adhesion. IEEE Transactions on Robotics, 24(1):65–74, Feb 2008.

[11] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth. A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. Advanced Engineering Informatics, 29(2):196 – 210, 2015. Infrastructure Computer Vision.

[12] G. Li, S. He, Y. Ju, and K. Du. Long-distance precision inspection method for bridge cracks with image processing. Automation in Construction, 41:83 – 95, 2014.

[13] J. J. Lim, C. L. Zitnick, and P. Dollar. Sketch tokens: A learned mid-level representation for contour and object detection. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 3158–3165, June 2013.

[14] N. Metni and T. Hamel. A uav for bridge inspection: Visual servoing control law with orientation limits. Automation in Construction, 17(1):3–10, 2007.

[15] F. M. Nejad and H. Zakeri. An optimum feature extraction method based on wavelet–radon transform and dynamic neural network for pavement distress classification. Expert Systems with Applications, 38(8):9442 – 9460, 2011.

[16] T. S. Nguyen, S. Begot, F. Duculty, and M. Avila. Free-form anisotropy: A new method for crack detection on pavement surface images. In 2011 18th IEEE International Conference on Image Processing, pages 1069–1072, Sept 2011.

[17] H. Oliveira and P. L. Correia. Automatic road crack detection and characterization. IEEE Transactions on Intelligent Transportation Systems, 14(1):155–168, March 2013.

[18] N. Otsu. A threshold selection method from gray-level histograms. Automatica, 11(285-296):23–27, 1975.

[19] M. D. Phung, C. H. Quach, D. T. Chu, N. Q. Nguyen, T. H. Dinh, and Q. P. Ha. Automatic interpretation of unordered point cloud data for uav navigation in construction. In 2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2016.

[20] P. Prasanna, K. J. Dana, N. Gucunski, B. B. Basily, H. M. La, R. S. Lim, and H. Parvardeh. Automated crack detection on concrete bridges. IEEE Transactions on Automation Science and Engineering, 13(2):591–599, April 2016.

[21] S. Rathinam, Z.W. Kim, and R. Sengupta. Vision-based monitoring of locally linear structures using an unmanned aerial vehicle. Journal of Infrastructure Systems, 14(1):52–63, 2008.

[22] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. Pattern recognition, 33(2):225–236, 2000.

[23] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen. Automatic road crack detection using random structured forests. IEEE Transactions on Intelligent Transportation Systems, 17(12):3434–3445, Dec 2016.

[24] S. K. Sinha and P. W. Fieguth. Segmentation of buried concrete pipe images. Automation in Construction, 15(1):47 – 57, 2006.

[25] P. Subirats, J. Dumoulin, V. Legeay, and D. Barba. Automation of pavement surface crack detection using the continuous wavelet transform. In 2006 International Conference on Image Processing, pages 3037–3040, Oct 2006.

[26] Y. H. Yu, N. M. Kwok, and Q. P. Ha. Color tracking for multiple robot control using a system-on-programmable-chip. Automation in Construction, 20:669 – 676, 2011.