Investigation of images of cracks via graph theory for developing an optimal exploration algorithm for a robotic manipulator

Francesca Palermo[1*], Changjae Oh[1], Kaspar Althoefer[1], Stefan Poslad[1], Ildar Farkhatdinov[1,2*]

***Abstract*— An important task often performed in remote hazardous environments is the detection of mechanical fractures on containers used for keeping chemical and radioactive waste. In this paper, an exploration algorithm for fracture localisation and inspection is proposed. Fracture images are inspected via computer vision processing techniques and a graph of the fracture is created. This graph is then inspected via a Minimum Spanning Tree algorithm and an optimal exploration path is defined. An experiment to validate the proposed algorithm is performed and it is found that the time required to explore a fracture using graph theory reduces the exploration time by a factor of $\sim$12 on average.**

I. INTRODUCTION

Detecting mechanical fractures on an object, such as containers and pipes, is an important task often performed in remote hazardous environments. In this situation, crack detection is particularly important since it can avoid spillage of hazardous material from the container or detect cracks on the surface of the concrete. Edge detection and image segmentation methods can be applied in supervised and well-structured environments when the crack has clear continuity and its image has high contrast [1]. However, cracks are usually found in noisy backgrounds which lead to poor continuity, low contrast and negatively impact the acquired imaging quality. Chen et al [2] proposed a fusion deep learning framework called NB-CNN (Naïve Bayes - Convolutional Neural Network). It analyses individual video frames for crack detection and detects crack patches in each video frame. Zou et al [3] proposed DeepCrack, a deep convolutional neural network for automatic crack detection. It recognises the line structures by employing multi-scale deep convolutional features learned at hierarchical convolutional stages. The above described crack detection methods are based on computer vision techniques and can fail in remote environments with limited luminosity or noise due to radiation. In contrast to the visual modality, tactile and proximity sensing can provide important information on material properties such as shape, texture and hardness [4]. In [5], [6] it is demonstrated that it is possible to use fiber optics to recognise and classify fractures on surfaces. For real-time applications, exploring the whole surface using only a tactile approach would be too time-consuming and may produce errors. Images of the crack can be acquired and further analysed to extract the geometry information and to plan an optimal path for tactile exploration. In this paper, we propose a vision-based algorithm that estimates an optimal exploration path for the robot to inspect the region of interests where the cracks are located.

II. SETUP

Previously, a tactile proximity algorithm for fracture detection was developed to localise possible fractures [6], [5]. The images of the crack can be acquired to be analysed by extracting the geometry information to plan an optimally controlled tactile exploration. To analyse the obtained localised crack images, image processing and computer vision techniques are implemented to create a skeletonised version of the image of the fracture which is then transformed into a graph and explored via graph theory. The key steps are demonstrated in Figure 1. First, to avoid obtaining open contours and incomplete masks, a uniform coloured padding is introduced in the acquired image (size 100×200 pixels) of the fracture (Figure 1a) to close potentially open locations. The colour of the padding is chosen as the average of the total RGB colours of the image. The original image is then converted to grey scale and blurred with a Gaussian filter (3x3 kernel), as shown in Figure 1b. The resulting image is converted to a binary image using Otsu and binary thresholds (Figure 1c). To connect the disconnected cracks in the estimated binary image, the dilation operation in morphological transformations is applied. Then, Canny Edge detection is implemented (Figure 1d). The average of the intensities of the pixels is used to automatically estimate the lower and higher threshold for edge detection. The resulting edges are improved with an additional dilation operation. Using the obtained edges, the contours of the fractures are calculated (Figure 1e). The averaged area of the contours is used to eliminate any outliers. The object mask is then created (Figure 1f), which is used to skeletonise the fracture (Figure 1g). For this purpose, we use an open-source PlantCV library[1] which provides an useful method to create a skeleton from the mask and to prune it. The *sknw* library[2] is applied to the resulting skeleton to create a first graph of the fracture with nodes and edges (Figure 1h). Middle points of the edges of the previously preliminary graph are extracted and an updated graph made up of these middle points is created with *Networkx* library[3] (Figure 1i). Each of these middle points is connected to the rest of the points

[1] The Centre for Advanced Robotics @ Queen Mary (ARQ), School of Electronic Engineering and Computer Science, Queen Mary University of London, London, United Kingdom

[2] Department of Bioengineering, Imperial College of Science, Technology and Medicine; London, United Kingdom.

$*$ {f.palermo,i.farkhatdinov}@qmul.ac.uk

[1] https://github.com/danforthcenter/plantcv
[2] https://github.com/Image-Py/sknw
[3] https://github.com/networkx/networkx

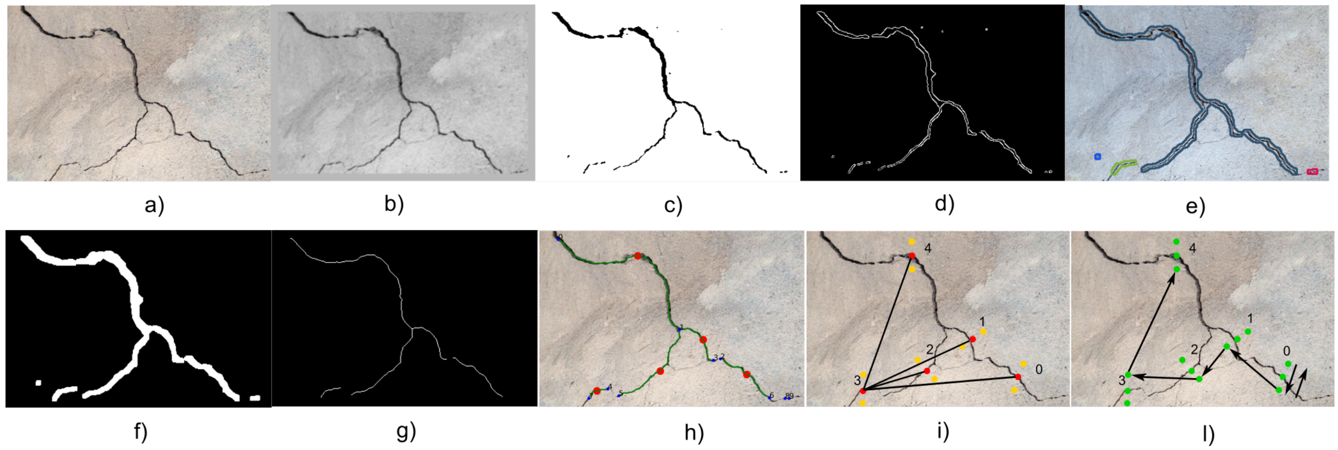

Fig. 1. Example of image processing for the crack's geometry analysis: a) original image; b) grey-scale and Gaussian blur; c) binary image; d) threshold Canny edge detection with morphological transformations; e) extraction of contours; f) binary mask g) pruned skeleton; h) graph and branches identified, in red the middle points of each edge is shown; i) the fully-connected middle points graph is created (for clarity, only the edges of node 3 are shown); l) the optimal exploration path is defined (only the node 0 left and right exploration are shown for brevity).

via Euclidean distance to create a fully connected graph. The distance is used to create the weighted edges of the graph. In addition, for each middle point, shifted left and right points are created which will be used to develop the motion plan of the manipulator with the fibre optic sensor, described in [6], attached as end-effector. These coordinates and weights can be used to define the tactile exploration path. Graph theory is then applied to investigate the possible exploration path (Figure 1l). Since the goal of the experiment is to explore all the possible nodes with the smallest cost (time), a revised version of the Minimum Spanning Tree is implemented. Each node of the graph and its left and right shift are explored only once and the minimum path from the starting node is found. As a result, a robotic-manipulator with a tactile sensor attached at the end-effector can directly explore only the main elements of complex cracks branch by branch following the paths identified with the help of geometrical analysis of the image of the cracks.

III. Data Analysis and Discussion

To validate the proposed scenario, an experiment to calculate the time required to analyse images with the graph exploration algorithm has been performed. A total of 25 images have been investigated and the average time required to explore them with both the graph theory and no graph has been extracted. Pictures with different height and width were used. Since in a realistic scenario the robot would move at a millimetres/second speed, the pixels have been converted in millimetres, using the dots Per Inch (DPI) value of the images. Considering that the majority of images had a DPI (dots per inch) resolution equal to 96, the conversion rate was equal to 0.26. For the analysis, one complete sliding from the left position to the right position with regards to the middle point node and backward was used. Two speeds were used in the experiment: one for exploring the interesting nodes and one to move from one node to the other. The speed used for exploring a node was the same as the average sliding velocity of [6], equal to 3.89 mm/s. This velocity is then doubled when moving from one node to the next. For the analysis with no graph exploration, since all the location in the image may contain useful information, only the speed for exploring a node was implemented. On average, analysing the fractures without the graph exploration, required 769 seconds ($\sim$12 minutes). The exploration time is reduced by $\sim$12 when the discussed graph theory is introduced, which requires on average 67.24 seconds ($\sim$1 minute).

IV. Conclusion and Future Work

This paper proposes a vision-based algorithm that estimates an optimal exploration path for the robot to inspect the region of interests where the cracks are located. A graph is created from the images of the fractures and then inspected via Minimum Spanning Tree algorithm to define an optimal exploration path. The experiment to validate the algorithm shows that the proposed method is $\sim$12 times faster than the baseline. As future work, we shall demonstrate how the proposed method is integrated with the motion planning of a robot for the tactile sensor to minimise the inspection time.

References

[1] A. Mohan and S. Poobal, "Crack detection using image processing: A critical review and analysis," *Alexandria Engineering Journal*, vol. 57, no. 2, pp. 787–798, 2018.

[2] F.-C. Chen and M. R. Jahanshahi, "Nb-cnn: deep learning-based crack detection using convolutional neural network and naïve bayes data fusion," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4392–4400, 2017.

[3] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, "Deepcrack: Learning hierarchical convolutional features for crack detection," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1498–1512, 2018.

[4] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Tactile sensing in dexterous robot hands," *Robotics and Autonomous Systems*, vol. 74, pp. 195–220, 2015.

[5] F. Palermo, J. Konstantinova, K. Althoefer, S. Poslad, and I. Farkhatdinov, "Implementing tactile and proximity sensing for crack detection," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 632–637.

[6] ——, "Automatic fracture characterization using tactile and proximity optical sensing," *Frontiers in Robotics and AI*, vol. 7, 2020.