

Pseudo-ground truth data collection on pavement images

V. Baltazart, L. Yang *, Ph. Nicolle, J-M. Moliard
Ifsttar, UBL University, Nantes, France

Abstract— The performance assessment of automatic crack detection algorithms within pavement images requires beforehand to establish a reference image, namely, the pseudo-ground truth image (PGT). In this context, this paper presents some existing pseudo-ground truth (PGT) data collection techniques which rely on image processing techniques. The processing of five Single Pair Shortest Path (SPSP) algorithms which are devoted to this aim are illustrated in terms of running time and segmentation accuracy on a pavement image.

Keywords—Road surface monitoring, crack detection, image processing, Single Pair Shortest Path, DICE similarity coefficient, performance assessment, pseudo-ground truth

I. INTRODUCTION

A lot of image processing techniques have been developed to computerize the survey of road surface conditions and for instance, to automatically detect the cracks within 2D pavement images, e.g., [1-6] and the references therein. However, none of the existing algorithms actually affords the ideal segmentation which may serve as the references for others. Then, the default solution for the performance assessment of the latter automatic algorithms compares the segmented results to some reference image at the suitable image scale, i.e., the pixel or any larger gridding. The reference image usually consists in a binarized representation of the crack pattern, namely, the pseudo-ground truth (PGT) image.

In this paper, we review some existing PGT data collection techniques with emphasis on those which require a semi-automatic (SA) process and then the skills of an operator. The SA process relies on image processing techniques. It requires initiating the segmentation by selecting either one or two endpoints within the visible crack pattern. In case of ramified crack structure, the operator breaks down the crack into elementary segments.

For the semi-automatic PGT data collection, this paper proposes two families of algorithms: percolation-based [7] and path finding algorithms, e.g., [2,8-11]. Each algorithm enables different amount of visited pixels, which in turn influences the computational time to collect the PGT. This paper focuses on the path finding algorithms which rely on the so-called Single Pair Shortest Path (SPSP) problem [8]. Dijkstra [9] and Fast Marching [10-11] are two widespread SPSP algorithms to this aim. The result on a pavement image illustrates the performance and the limitations of the cited algorithm families in terms of segmentation accuracy and computational time. Finally, the paper addresses the condition to use the automatic crack detection algorithms for faster PGT data collection.

II. EXISTING PROCESSES FOR PGT DATA COLLECTION

A. Manual PGT data collection

From an historical point of view, the operators performed the pavement survey by the visual inspection of the pavement surface at walking speed. The detected pavement distresses were then notified and/or recorded along some length. Today, electronic device and the GPS technology may provide a virtual gridding of the road surface that ease the record of pavement distresses at walking speed in order to establish some statistics.

With the use of imaging devices which record pavement images at traffic speed, in situ visual inspection at walking speed was replaced by the visual inspection of the recorded images on the screen. The GT data collection then became the pseudo-ground truth (PGT) because of the loss of direct inference with the in situ information.

Then, the operators spend much time on the screen to collect the pavement distresses. The PGT data collection is performed on some image blocks (larger in size than the pixel gridding for obvious reasons), which the size is matched to the existing operational survey method in each country. Some image processing tools, e.g., CrackIT in [1], include user interface to collect the PGT on some image gridding. For example, the PGT is collected on 20×20 cm² gridding in [5]. A specific coding for each grid allow to record different kinds of pavement distresses for later use. The average density of defective grids per square meters over 50 m is then used as a survey indicator. In [12], the pavement distresses on each image are aggregated over 200 m, in order to detect the structural pathologies with large longitudinal extension.

The manual PGT data collection at the pixel level has been used in [6] for research purpose. Using a simple graphical user interface, four operators manually selected on the screen the pixels belonging to the crack pattern. 36 pavement images from the previous French imaging device, namely, Aigle-RN, were then manually segmented. The union of the four segmented results formed the PGT on the pixel gridding which was used for the benchmarking of some crack detection algorithms. The manual PGT was proved later in [2] to be less accurate than the semi-automatic PVT, which is presented in the next section.

In any case, the data collection of pavement distresses is a time consuming and very laboring task. The amount of images to analyze may depend on the pavement distresses to detect. As a consequence, a very few image data base exist in the literature that includes the PGT. Besides, this prevents

(*) actually MS student at Xi'an Jiaotong University, Shaanxi, China.

performing the rigorous assessment of the existing crack segmentation algorithms.

B. Semi-Automatic PGT data collection

In the semi-automatic method, the operator interactively selects on the screen either one or two endpoints within the crack pattern, namely, a source and/or a destination pixel. Then some algorithms automatically determine the skeleton of the crack segment. The crack pattern is thus segmented in successive segments according to its geometry as shown in Fig. 1. Nonetheless, some algorithms in section III may show the capability to segment the ramified structure of the crack pattern at some extent.

At last, the crack width is estimated along the crack skeleton, by iteratively aggregating the pixels which the grey level is lower than the following threshold [2]:

$$T_w = \mu_w - \kappa_w \sigma_w \quad (5)$$

where μ_w and σ_w are the mean and the standard deviation of the grey level distribution of the crack pixels, respectively. This step conventionally assumes the intuitive relationships between lower grey level pixels and the cracking. The parameter κ_w has been fixed in this work to 0.6, accordingly to the value which was used in [2] for optimizing the automatic crack detection algorithm MPS.

In [13], different family algorithms have been tested to fill this purpose. Most of them are based on path finding techniques to determine the pieces of the crack skeleton between two endpoints. By contrast, the percolation algorithm in [7] was appealing because it requires initiating the path search from one endpoint only. The percolation method has been successfully tested on concrete images but was proved in [13] to achieve poor result on pavement images owing to the stronger image texture. Besides, it was found slower than any of the tested path finding algorithms. Then, the latter family of algorithms has been selected in the next section for the semi-automatic segmentation of the crack pattern.

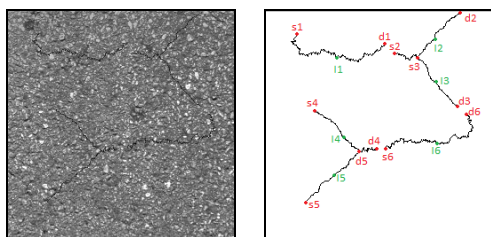


Fig. 1 Synthetic pavement image used for the test; on the left, synthetic image in [2]; on the right, associated ground truth; in red, the 6 pairs of selected sources and destination seed points for SPSP path search algorithms; in green, the 6 seed points used for initializing the path search by ST Fast-Marching.

III. PATH FINDING TECHNIQUES FOR SEMI-AUTOMATIC PGT DATA COLLECTION

Two families of SPSP algorithms have been tested in this section. The Dijkstra's algorithm [9] is a ubiquitous algorithm which can be used indifferently for the one-to-one (or

equivalently, single pair shortest path or SPSP) and one-to-all (or single source shortest path or SSSP) problems. It minimizes the Manhattan distance between two pixels of the image under the L1 standard norm. In comparison, the other tested algorithms are dedicated to the SPSP problem only.

In the bidirectional version of Dijkstra, the search for the shortest path is initialized simultaneously at both the source and the destination endpoints. Then, the path finding algorithm proceeds until a junction endpoint is found. The number of visited pixels is thus reduced by about half, as well as the execution time on each crack segment.

The * (star) family algorithms have been proposed as an alternative to Dijkstra. A * is known as the fastest and the most accurate algorithm in the latter family. A * uses a heuristic function, which directs the path search between the ongoing pixel and the destination pixel. It has been adapted in [8] to the PGT data collection within pavement images.

By contrast, the Fast Marching method [10-11] is a SPSP algorithm, which minimizes the geodesic distance between two pixels using the standard euclidean L2 norm. It has been widely used in image processing for pattern recognition, e.g., active contours.

A recent variant of the Fast Marching technique in [14], which is labelled Self Terminating Fast Marching (STFM) in this paper, requires initializing the path finding at one endpoint only, the other endpoint of the crack being automatically determined by some stopping criteria. The path finding can propagate to the branches of a crack at some extent. This may facilitate the work of the operator to determine the PGT. The counterpart is a longer execution time owing to greater computational complexity.

IV. ASSESSMENT PROTOCOL

The performance assessment of the semi-automatic segmentation of the cracking relies on the computation of the conventional similarity indices between the segmented results by the SPSP algorithms and the reference segmentation in the data base [15]. For field pavement images, the latter reference segmentation has been previously established in [2] by a single operator using the Dijkstra algorithm. For synthetic images, the crack pattern is user-defined and then is qualified as ground truth.

The evaluation is performed on the pixel gridding to obtain the most precise and sensitive analysis. We use a direct comparison between the PGT and the segmented result; for comparison, a 2 pixels margin was used in [2] and a larger margin in [3]. The performance evaluation involves the following four categories of pixels: true positive pixels (TP), false positive pixels (FP), false negative pixels (FN) and true negative pixels (TN).

The three first categories of pixels, i.e., TP, FP and FN pixels, are used to compute the precision (P) and the sensitivity (S or recall) which indicates the rates of false alarm and the rate of undetected crack pixels, respectively. Finally, the harmonic average of the two latter coefficients is called the Dice-Sorensen coefficient (DSC) or the F1-score, equivalently,

and is conventionally used for performance assessment in image processing. In practice, the three criteria are 1 if the two images are perfectly matched to each other, and 0 if there is no superposition. A large difference between the S and P coefficients strongly impacts the DSC rate. The segmentation result is usually considered as good enough for DSC value above 0.7 [16].

In addition, the efficiency of the different SPSP algorithms was recorded through the computation time and the number of pixels that the algorithms visited for the processing. The latter is called the query area and is displayed in grey on Fig. 2. The five SPSP algorithms were coded with the Matlab programming language and were tested on the same personal computer [13]. Besides, each crack pattern is decomposed using the same crack ramification (see Fig. 1 for the illustration on the synthetic crack pattern), and the starting and the destination pixels of each ramification are the same to perform the semi-automatic crack detection by the different algorithms.

V. ILLUSTRATIVE EXAMPLES

The five SPSP algorithms have been tested on both simulated and pavement image samples. Both image types are from the data base in [15], which also includes the pseudo ground truth.

A. Test on a pavement image sample

The ranking of the five tested algorithms have been found identical on both image types [13]. In Fig. 2, the five

algorithms are performed on the pavement image 462×690 pixels in size, which has been used in [2-4] for testing automatic crack detection algorithms. The black line represents the crack (with the thickness accounted for) and the query area in grey displays the visited pixels by the algorithms

The results shown in Fig. 2 are ranked from the fastest to the slowest algorithm. To give an order of magnitude, the A* method, namely, the fastest one, takes about 4 sec on the HP Zbook14 Intel Core i7 at 2.7 GHz. The STFM method evaluates the second lowest number of pixels but takes about 60 sec because of greater computational complexity.

All algorithms provide very good segmentation with DSC value above 0.85, as depicted in the last column on Fig. 2. The ranking of the method is then found to be Dijkstra (0.94), A* (0.92), ST Fast Marching (0.87) and Fast Marching (0.86). Nonetheless, the three algorithms which minimize the L1 norm, namely, Dijkstra, bidirectional Dijkstra and A*, achieve larger DSC by a few percent, meaning that they better match to the chaotic geometry of the cracks. By contrast, the L2 techniques, i.e., FM and STFM, tend to smooth the segmented crack in shape.

For the three L1-based algorithms, there is a clear relationship between the size of the query area and the increasing computational time. Then, compared with Dijkstra's algorithm, the proposed A* is about 4 times faster and achieves equivalent DSC rate. Thus, the latter algorithm has been selected as the best trade-off algorithm for future PGT data collection.

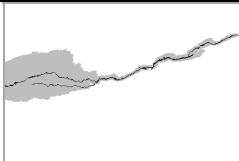



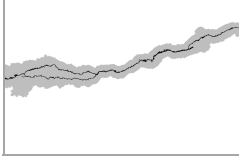
Algorithm	Query area	Relative amount of visited pixels	Relative computational time	DICE Similarity coefficient
A*		12.0 %	×1	0.92
Bidirectional Dijkstra		34.4 %	× 1.6	0.94
Dijkstra		61.4 %	× 4	0.94
Fast Marching		63.7 %	× 6.6	0.86
Self-Terminating Fast-Marching		12.5 %	× 14.8	0.87

Fig. 2. Semi-automatic segmentation obtained by the 5 tested SPSP algorithms on a 462×690 pavement image sample (from the fastest to the slowest).

B. Shortcomings of SPSP algorithms

The tests of SPSP algorithms on pavement images have highlighted two kinds of shortcomings which may affect the PGT data collection.

1) L1. vs. L2 norm

The two family algorithms, namely, Dijkstra-based and Fast-Marching-based algorithms, rely on L1 and L2 norms, respectively. These norms may lead to slightly different segmented results in practice. The latter difference has been theoretically illustrated in [17]. In this section, this difference is illustrated on pavement images.

When considering a pavement image with large cracks as shown on Fig. 3, we must be aware that the crack skeleton under the L1-norm does not match to the topological centerline of the crack as opposed to the result by the L2-norm. According to the taxicab geometry (or city-block distance), the shortest path under the L1 norm may not be unique within a uniform image background. In most cases however, cracks with large thickness depict a narrow grey level distribution; the resulting small grey level variations insures a unique shortest path to be determined under the L1 norm.

Finally, the thickness estimation of the crack at the final stage of the segmentation algorithm (see section II-B) allows mitigating the latter difference. L1 and L2 norms are thus shown to achieve very close final segmentation with a small difference on the DSC rate [2]. Besides, the crack width is likely to be thinner (in pixels) on pavement images than on the concrete image in Fig. 3.

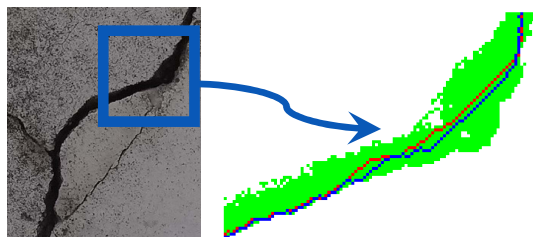


Fig. 3. L1 vs. L2 shortest path within a large crack on concrete image: Dijkstra in blue, Fast Marching in red, cracking pattern in green.

2) Shortcuts

The second shortcoming is illustrated on Figs. 4 for the cases of both thin cracks and ramified long cracks. In case of thin cracks, L1 and L2 SPSP algorithms may locally provide different segmented results, as illustrated within the circle in Fig. 4 (left). Dijkstra-based algorithms (in blue) provide the stair-step shortest path cost along the pixel gridding of the image that matches to the PGT (in green). By contrast, Fast-Marching (in red) may provide local shortcut within the image texture with smoother crack pattern. This implies a few percent differences on the DSC rate between the two family algorithms.

When the two endpoints take part on different branches of the crack as displayed on Fig. 4 (right), the tested path finding algorithms provide a solution through the image background

(in red) instead of following the crack pattern (in dark grey). STFM has shown to mitigate this discrepancy at some limited extent. The only solution for now to improve the PGT data collection in that case has been to consider the crack ramification (on the right) as an additional endpoint. Within this scope, interactive tools, e.g., [18-20], may allow reducing the amount of required endpoints and thus may help the user to collect the PGT. An alternative path search strategy, e.g., [21], may be also tested within a more general framework.

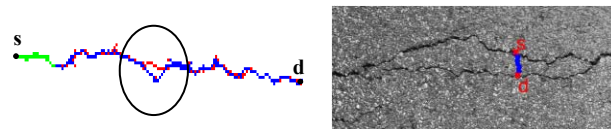


Fig. 4. Illustration of shortcuts: (left) case of thin chaotic crack: the Fast Marching method (in red) provides locally a shortcut (within the circle) compared to the Dijkstra's result (in blue); the PGT of the cracking pattern is displayed in green for visual reference; (right) case of ramified crack: the shortest path between (s) and (d) goes throughout the image background

VI. DISCUSSION

A. Reliability of the assessment protocol

The assessment protocol in section III is based on the comparison at the pixel scale between the semi-automatic segmentation and the reference segmentation which is available from [15].

Then, for field pavement images, it can be argued that the assessment protocol provides a bias, because the same family algorithm, namely Dijkstra, is used to provide both the reference PGT data collection and the computed semi-automatic segmented results. The latter ambiguity can be removed because the ranking of the five tested SPSP algorithms have been found identical on both synthetic and field pavement images [8,13].

In practice, the performance assessment of algorithms depends at a larger proportion on the quality of the reference image, and the skills of the operator to this aim.

B. Top and floor DSC rates

The DSC coefficient has been conventionally used in this paper to measure the quality of the semi-automatic segmentation. The image community is attached to the accuracy of segmentation results and will promote the algorithms which achieve the highest DSC rate. Within this scope, the DSC rate is computed on the pixel gridding.

In practice, DSC rate beyond 70% is considered as a good segmentation result [16]. The latter floor rate may vary from one application to one another. It especially depends on the contrast between the crack features and the image background, i.e., the image texture. Then, one can imagine considering a higher DSC floor rate for images with high contrast and a lower one otherwise. To our knowledge, such guidelines are missing at the operational level to assess the segmentation on the image gridding larger than pixel.

According to section IV, it can be argued that the top DSC rate is not the expected 100%, meaning that the ideal segmentation cannot be achieved. In fact, the reference PGT in [15] has not been performed by the same people at first. Second, the parameter κ_w in eqn. (1) was set to the value, which has been optimized in [2] for the automatic crack detection. Nonetheless, for the latter imperfect parameter setting, the top DSC rate has been found to be 90% on the pixel gridding.

Many other parameters may influence the top and the floor DSC rates in practice. Among others, it is shown in [2] that the DSC value is very sensitive to the pixels margin which is used to compute the DSC rate. The very large pixel margin in [3] artificially increases the DSC rate (up to 90%) but strongly reduces the difference between methods to a few percent as a counterpart, making hazardous the selection of the best segmentation method. By contrast, no margin is considered in this paper to compute the DSC rate.

VII. CONCLUSION AND PERSPECTIVES

This paper has reviewed some existing PGT data collection techniques and proposed some semi-automatic techniques which are based on image processing tools. Shortest paths algorithms have been found better matched to the PGT data collection in terms of usefulness, segmentation performance (DSC rate) and computational time. Most of them require the user to select two endpoints per crack ramifications. Among the five tested shortest path algorithms, A* has been selected as the best trade-off between segmentation performance and computational time.

The proposed semi-automatic processing is expected to help the users to collect the PGT on pavement images, and as a result, to favor the comparison between automatic crack detection algorithms. Despite the proposed improvements in this paper, the PGT data collection would remain in practice a laboring and time-consuming task, which the quality depends on the skills and the carefulness of the operator.

Finally, a faster two-steps PGT data collection is proposed as a working perspective. To this aim, the operator is asked to refine the segmented result which is achieved beforehand by some automatic crack algorithm. By superimposing the latter first segmentation on the ongoing pavement image, the user may complete and/or refine the PGT by performing a semi-automatic segmentation at appropriate locations on the pavement image. The latter solution may be worth pending further progress on automatic crack detection algorithms. The ongoing improvements in [22-23] on MPS do not achieve yet the top DSC rate (discussed in section VI-B) to refrain from the PGT data collection.

REFERENCES

- [1] ICIP 2014 Special session 19, "Image Processing for the Detection of Road-Surface Degradations" chaired by P.L. Correia and H. Oliveira, IEEE Int Conf. on Image Processing (ICIP), Paris, France, 2014
- [2] R. Amhaz, S. Chambon, J. Idier, V. Baltazart, Automatic crack detection on 2D pavement images : An algorithm based on minimal path selection, IEEE TITS, Vol 17(10), pp 2718-2729, 2016
- [3] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, Automatic Road Crack Detection Using Random Structured Forests, IEEE TITS, Vol. 17(12), 2016
- [4] Oliveira, H.; Correia, P.L.; "Automatic Road Crack Detection and Characterization", IEEE Trans. on Intelligent Transportation Systems, Vol. 14, No. 1, pp. 155 - 168, March, 2013.
- [5] Wright D., V. Baltazart, N. Elsworth, R. Hamrouche, J. Karup, M. Lurdes Antunes, S. McRobbie, V. Mereco and T. Saarenketo, D4.3 Monitoring structural and surface conditions, Tomorrow's Road Infrastructure Monitoring and Management (TRIMM), FP7 project 285119, 2014.
- [6] Chambon S. and J-M. Moliard, Automatic Road Pavement Assessment with Image Processing: Review and Comparaison, International Journal of Geophysics, pp. 1-20, 2011.
- [7] Yamaguchi T., S. Nakamura and S. Hashimoto, An Efficient Crack Detection Method Using Percolation-Based Image Processing, ICIEA 2008.
- [8] L. Yang, V. Baltazart, R. Amhaz, P. Jiang, A new A-star algorithm adapted to the semi-automatic detection of cracks within grey level pavement images, 8th ICDIP congress, Chengdu, China, 2016
- [9] E. W. Dijkstra, "A note on two problems in connexion with graphs," in *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.
- [10] L.D. Cohen and Ron Kimmel, "Global minimum for active contour models: A minimal path approach," *International Journal of Computer Vision*, vol. 24, no. 1, pp. 57–78, 1996
- [11] Charbonnier P., J-M. Moliard, Calculs de chemins minimaux, suivi de fissures et autres applications, in LCPC, editor, Journées des Sciences de l'Ingénieur du réseau des laboratoires des Ponts et Chaussées, pp 201–206, Dourdan, France, décembre 2003.
- [12] Bertrand L., Ph. Lepert, Relevé des dégradations de surface des chaussées, Testing method LPC 38-2, LCPC, 1997
- [13] L. Yang, Contributions to improve and speed the existing MPS-based algorithm for both the semi-automated and the automated crack segmentation on 2D pavement images, internship report, LUNAM University, France, 2015
- [14] Kaul V., A. Yezzi, and Y. Tsai, "Detecting curves with unknown endpoints and arbitrary topology using minimal paths," in *IEEE Trans. PAMI*, vol. 34, pp. 1952–1965, 2012.
- [15] Aigle-RN, Cerema, DTer Ouest, http://www.ouest.cerema.fr/IMG/pdf/120925_route-AigleRN_cle78acd3.pdf, accessed on 18 January 2016
- [16] Sampat M.P., Z. Wang, S. Gupta, A.C. Bovik and M.K. Markey. Complex Wavelet Structural Similarity : A New Image Similarity Index. *IEEE Transaction on Image Processing*, 18(11):2385–2401, 2009.
- [17] Deschamps T., Extraction de Courbes et Surfaces par Méthodes de Chemins Minimaux et Ensembles de Niveaux. Applications en Imagerie Médicale 3D, Thèse de l'Université Paris Dauphine - Paris IX, 2001.
- [18] Falcao, A. X, J. K. Udupa, S. Samarasekera, S. Sharma, B. E. Hirsch and R. A. Lotufo, User-steered image segmentation paradigms: live wire and live lane, *Graphical models and image processing*, 60, 223-260, 1998
- [19] Gérard O., T. Deschamps, M. Greff, and Laurent D. Cohen, Real-time Interactive Path Extraction with On-The-Fly Adaptation of the External Forces, *Computer Vision — ECCV 2002, Lecture Notes in Computer Science Volume 2352*, pp 807-821, 2002
- [20] Mortensen E. N. and W. E. Barrett, Interactive segmentation using intelligent scissors, *Graphical Models and Image Processing*, 1998
- [21] N. Zhu and A. C.S. Chung, Minimum Average-Cost Path for Real Time 3D Coronary Artery Segmentation of CT Images, *Proceedings of MICCAI 2011, Part III*, pp. 436–444, 2011.
- [22] V. Baltazart, Ph. Nicolle, L. Yang, Ongoing tests and improvements of the MPS algorithm for the automatic crack detection within grey level pavement images, *Eusipco*, Kos Island, Greece, 2017.
- [23] W. Kaddah, M. Elbouz, Y. Ouerhani, V. Baltazart, A. Alfalou, Optimized MPS for the automatic and unsupervised crack segmentation within two-dimensional pavement images, unpublished, 2017.