AI-Driven Matching of Pavement Surface Images Captured by In-Vehicle Smartphones

Krishna Modak
Dept. of Civil Engineering, The University of Tokyo; email: k.modak@bridge.t.u-tokyo.ac.jp

Kai Xue
Smart City Research Institute; email: kai.xue@smc-tech.com

Di Su
Dept. of Civil Engineering, The University of Tokyo; email: su@bridge.t.u-tokyo.ac.jp

ABSTRACT

Pavement monitoring is essential for ensuring the safety, efficiency, and longevity of transportation infrastructure. By regularly assessing pavement conditions, authorities can identify hazards such as cracks, rutting and potholes, enabling timely repairs to prevent accidents and injuries. Moreover, monitoring allows for the optimization of resources, ensuring that maintenance efforts are targeted where they are most needed, thereby extending the lifespan of pavements and reducing long-term costs. Non-destructive testing methods valuable data without harming the pavement structure, though they may vary in complexity, cost, and the depth of information they provide. Imaging technology has been widely applied in pavement detection field, there are several types of pavement inspection devices for different pavement distress, for example, digital camera to detect the pavement crack, rutting and potholes. This study presents the chronological evolution of pavement surface images got by a smartphone over a three-year duration. Initially, video footage of the road surface is recorded using a smartphone securely affixed to a vehicle's interior windshield at an oblique angle. Given that the recorded video is not captured from a top-down perspective, its distortion gives it suboptimal for analysis; thus, it is transformed into a bird's-eye view. Subsequently, these images undergo sorting and alignment based on GPS segments. However, it is noted that sorting by GPS may lack precision due to inherent errors and variations in vehicle speed. Each measurement yields its sequence of images, making exhaustive comparison computationally impractical. A method for finding potential pairs for each image is developed to solve this. This involves a preliminary screening of a select subset of images, which are then matched utilizing AI-based feature matching. A stitching procedure is implemented upon identifying the best pairs from two distinct measurements. Following this method, variations in pavement surfaces, such as the proliferation of crack intersections, enlargement of potholes, emergence of new potholes, or instances of repair work, are observed.

KEYWORDS: Structural health monitoring; Nondestructive testing; Smartphone; Image; AI.

1. INTRODUCTION

Pavement monitoring is crucial for ensuring safety, economic efficiency, and the longevity of infrastructure. By regularly assessing pavement conditions, hazardous issues such as potholes, cracks, or uneven surfaces can be identified and addressed promptly, preventing accidents and enhancing road safety. Economically, early detection of problems allows for cost-effective maintenance, optimizing resource allocation and avoiding expensive major repairs or complete reconstructions. This proactive approach significantly extends the lifespan of pavements by addressing minor issues before they escalate. It also ensures a smooth and comfortable driving experience, reducing vehicle wear and tear and minimizing noise pollution for nearby residents. Environmentally, routine monitoring promotes sustainability by reducing the need for extensive reconstruction projects that consume considerable resources and energy. The data gathered from monitoring aids in informed planning and decision-making, tracking pavement performance, and improving construction practices. Moreover, it ensures compliance with regulatory standards and maintains the quality of the infrastructure.

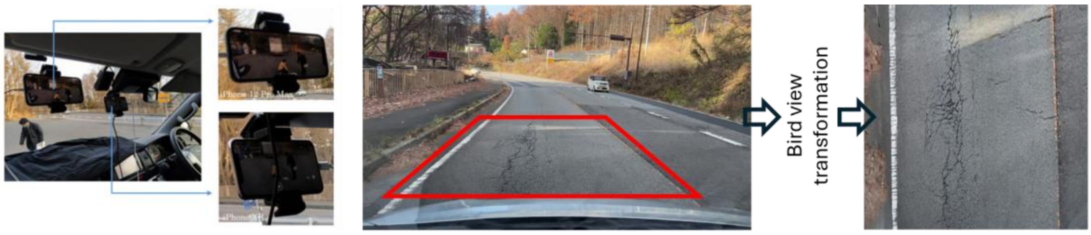
Monitoring pavements involves various methods tailored to assess their condition, performance, and structural integrity. Visual inspections conducted by trained personnel remain a fundamental approach, allowing for the identification of surface distress such as cracks and potholes. Destructive testing methods like coring, plate load testing, and core drilling offer accurate insights into pavement properties and structural performance but are invasive, time-consuming, and may cause pavement damage. Non-destructive testing methods such as ground-penetrating radar (GPR) (Benedetto (2007)), falling weight deflectometer (FWD) (Mehta (2003)), light weight deflectometer (Umashankar (2016)), and visual inspection provide valuable data without harming the pavement structure, though they may vary in complexity, cost, and the depth of information they provide. GPR and FWD offer comprehensive subsurface data but require specialized equipment and expertise, while visual inspection is simple and cost-effective but may lack depth and subjectivity in the analysis.

In recent years, in-vehicle smartphones have been developed as an alternative for monitoring pavement surfaces. Utilizing various sensors such as IMUs, GPS, and components like smartphone cameras, indices for monitoring, including IRI, rutting, and potholes, can be assessed. The main advantages of using in-vehicle smartphones as an alternative are ease of monitoring and data acquisition, cost-effectiveness, and speed, albeit with less accuracy than conventional methods. Gao (2023) utilized in-vehicle smartphone images and image processing to estimate rut depth. Xue (2020) employed a half-car model and vibrational data from smartphones to estimate the IRI. Kyriakou (2019) used inertial data from smartphone sensors and ANN to develop a pothole detection model. The frequent availability of data allows for the observation of changes in pavement surfaces.

In this study, smartphone imagery obtained from in-vehicle is employed to monitor a specific section of pavement over three years. Monitoring is conducted biannually, resulting in a total of six measurements. Each measurement yields over 550 images, though the number varies between measurements. These images must be efficiently matched with subsequent measurements to identify the best matches without excessive time and computational resources. This paper presents the various methods and tools utilized to achieve optimal image pairing.

2. MEASUREMENT SETUP AND PRE-PROCESSING

To conduct the measurements and obtain the images, two items are required: a car and a smartphone. The smartphone is positioned on the windshield, as shown in Figure 1(a). The car is then driven over the designated section while recording video of the road surface. The recorded video is decomposed into frames to obtain the images. These images are not initially in the appropriate perspective for observing chronological changes as seen in Figure 1(b). Consequently, they are transformed into a top-view perspective using bird's-eye transformation, as illustrated in Figure 1(c). This process is repeated for each measurement conducted.



(a) Measurement setup (b) Captured video frame (c) Required perspective

Figure 1 Smartphone positioned on the windshield for measurements.

3. SEARCHING BEST PAIR MATCH

After obtaining the bird's-eye view images, it is necessary to match all possible images from previous measurements to the current measurement. Two issues arise: a) the number of images can vary for each measurement as the starting and ending times are not consistent, and b) even if the starting and ending points of the section are recorded accurately, GPS errors cause the images to be unsynchronized. Therefore, all images must be included to search for the best matching pairs. The details of the measurements conducted are summarized in Table 1.

Table 1 Measurement Details

Measurement	Date	Time	Smartphone
1	2021.03.11	17:01	iPhone 12 Pro Max
2	2021.11.30	13:48	iPhone 13 Pro
3	2022.03.15	17:02	iPhone 13 Pro
4	2022.11.25	15:19	iPhone 14 Pro
5	2023.04.19	16:17	iPhone 13 Pro
6	2023.11.27	14:56	iPhone 13 Pro

To match the images, an AI-based feature-matching algorithm, LightGlue (2023), is utilized. LightGlue is a deep neural network designed for efficient and accurate local features matching across images. It improves upon the previous state-of-the-art method, SuperGlue, offering greater memory and computational efficiency, as well as enhanced accuracy. It is important to note that all measurements were conducted under varying weather and sunlight conditions (refer to Figure 2), which may affect matching efficacy. The best match pair is identified by the highest number of features matched between two image pairs compared to other pairs. For example, Figure 3 demonstrates the comparison and matching of two images from the same measurement day and two images from different measurement days. It is evident that same-day measurements exhibit more feature matches than measurements from different days. The search methods discussed in the below subsections were implemented on Nvidia 1080ti GPU device.

In the following subsections, we detail the various methods employed to obtain the best match pairs.

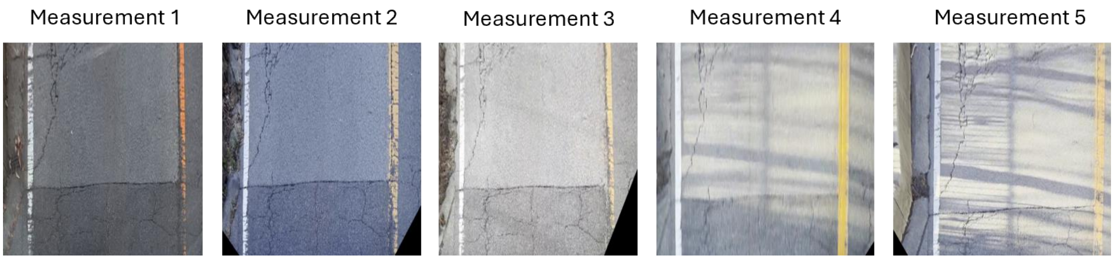
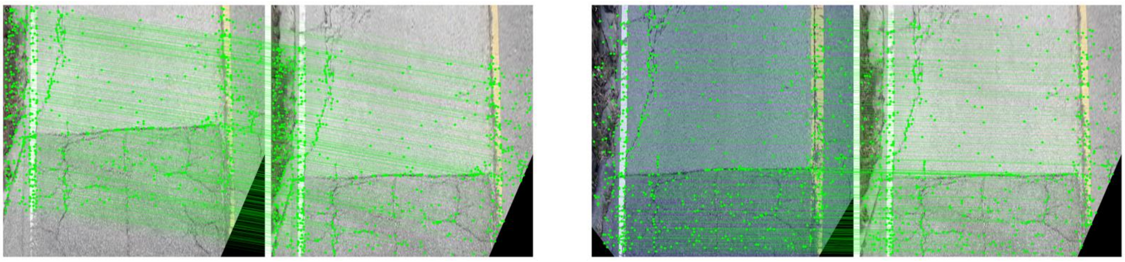


Figure 2 Factors influencing the matching: weather, lighting conditions, shadows, blurring, sun position



(a) Feature matching for same-day measurement

(b) Feature matching for different day measurement

Figure 3 Comparison between same-day and different days feature matching.

3.1. *All images of past measurement compared with all current measurement images*

This method, while basic, is highly time-consuming for finding the best image pairs. Each image from past measurements is compared with every image from the current measurements. After identifying the best match, the paired images are stored, and the process is repeated for the next image from the past measurements. This continues until all images from past measurements have been compared with all images from the current measurements. Figure 4 illustrates the search process. In this example, the first measurement contains 22 images, and the second measurement contains 17 images for the same segment length. Due to GPS errors, the images are out of sync, meaning the first image of the first measurement does not correspond with the first image of the second measurement. After the search, the image indices from the first and second measurements with the highest feature match are stored. For instance, the fourth image of the first measurement might match with the first image of the second measurement, and so on. This entire search process of comparing 22 images to 17 images took 1 minute and 43 seconds. We refer this search as basic search.

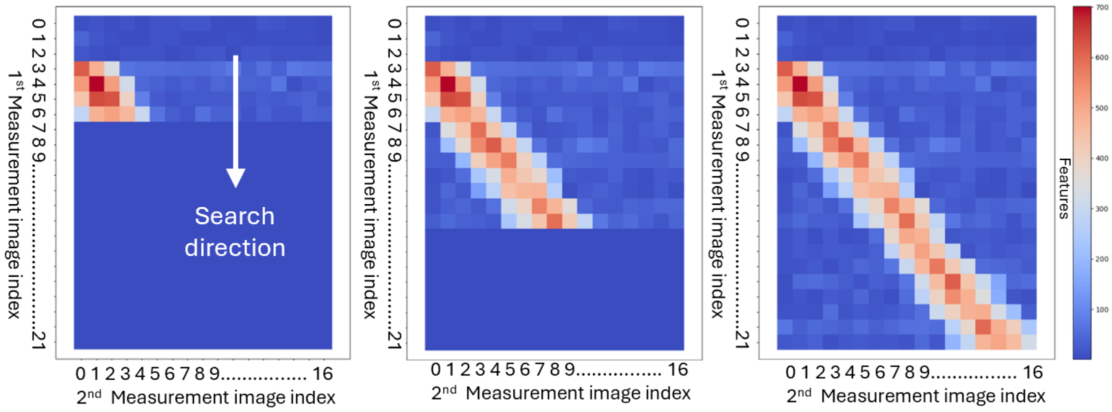


Figure 4 All images of 1st measurement were compared with all images of 2nd measurement.

3.2. *Searching the best pair diagonally*

This search method assumes that the first image from the previous measurement is in the vicinity of the first image of the current measurement due to GPS errors. The algorithm searches a few images before and after the current image (search window)(Figure 5(b)), thereby avoiding the need to examine all images and saving time and computational power. The best match among these few images is recorded. Figure 5 illustrates the implementation of this search method with the same example as in Section 3.1. We refer this search method as diagonal search.

However, this algorithm has critical shortcomings. Since the vehicle speed varied across different measurements, the image number offset is not constant throughout the measurements, compounded by GPS errors. This results in a drift from the diagonal. Also the size of window search is uncertain before search. Additionally, if the search window is not large enough, the algorithm may fail to identify the best-matched pair, as shown in the bottom right corner of the last plot in Figure 5(c). This entire search process of comparing 22 images to 17 images search took 1 minute and 3 seconds. It is also evident that searching within windows (7 images in the vicinity) on both sides of the diagonal did not prove useful, further consuming additional time and computational resources. This indicates that there is a more efficient method for finding the best image pairs.

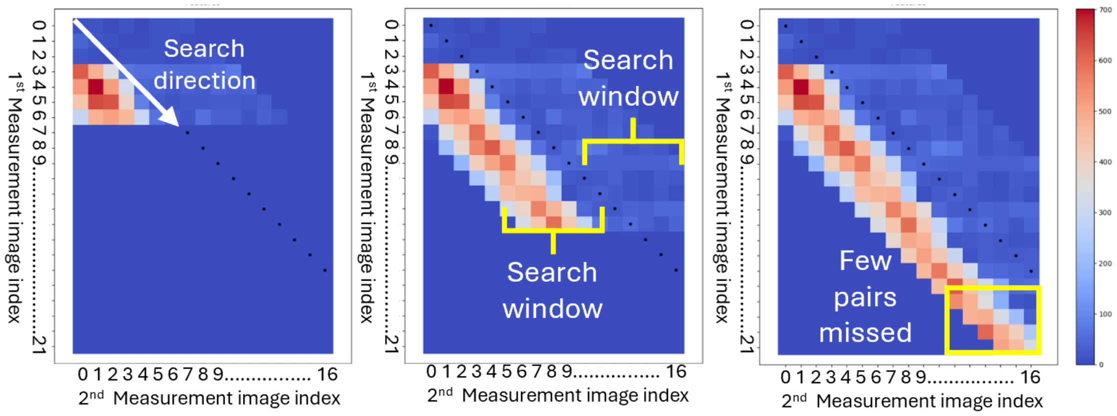


Figure 5 Searching best match pair diagonally.

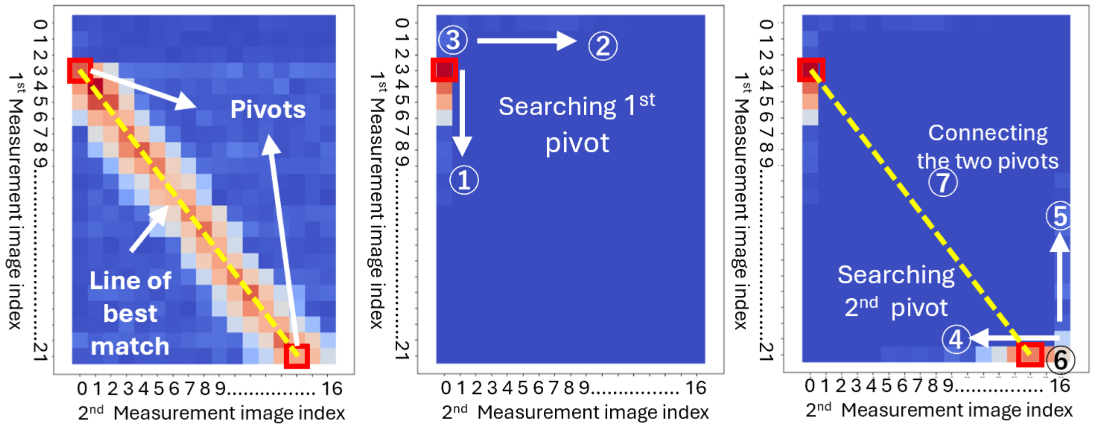
3.3. *Searching along a most probable best match line*

In this search algorithm, we aim to identify a line that passes through all the best matches, as illustrated in Figure 6.a. To establish this line, two points or pivots are required. A pivot is the best match image pair at the start or end of a measurement sequence. Specifically, a pivot is the sequence number or index of the starting or ending images of the two measurements. Below explains the search of the line:

① Search for the best match for 0^{th} index image in 1^{st} measurement in 2^{nd} measurement.
② Search for the best match for 0^{th} index image in 2^{nd} measurement in 1^{st} measurement.
③ Register the best match among in step ① and ②. This is the first pivot for the line.
④ Search for the best match for n^{th} index image in 1^{st} measurement in 2^{nd} measurement.
⑤ Search for the best match for m^{th} index image in 2^{nd} measurement in 1^{st} measurement.
⑥ Register the best match among in step ④ and ⑤. This is the second pivot for the line.
⑦ Connecting the two pivots gives the line of best match.

Where n & m are the number of images in 1^{st} and 2^{nd} measurement respectively.

This search algorithm begins by identifying the two pivots to create a line. One pivot is located in the top left corner and the other in the bottom right, as shown in Figure 6(b) & 6(c). After identifying the pivots, a line connecting them is drawn (⑦). This line represents the most probable best match line. The indices along this line are considered the most probable best matches. To enhance robustness, a small search window along the line is implemented, similar to the method in Section 3.2. We refer this search as line search. Figure 7 demonstrates the completion of the search. Compared to the algorithms described in previous sections, this algorithm requires less time and ensures the identification of all best-match pairs. Using the same example as in Section 3.2, this search took only 33 seconds, without missing any best match pairs, with a small search window (1 search in the vicinity).



(a) Pivots and line of best match

(b) Searching first pivot

(c) Searching second pivot and line of best match

Figure 6 Steps for searching the pivots and the line of best match.

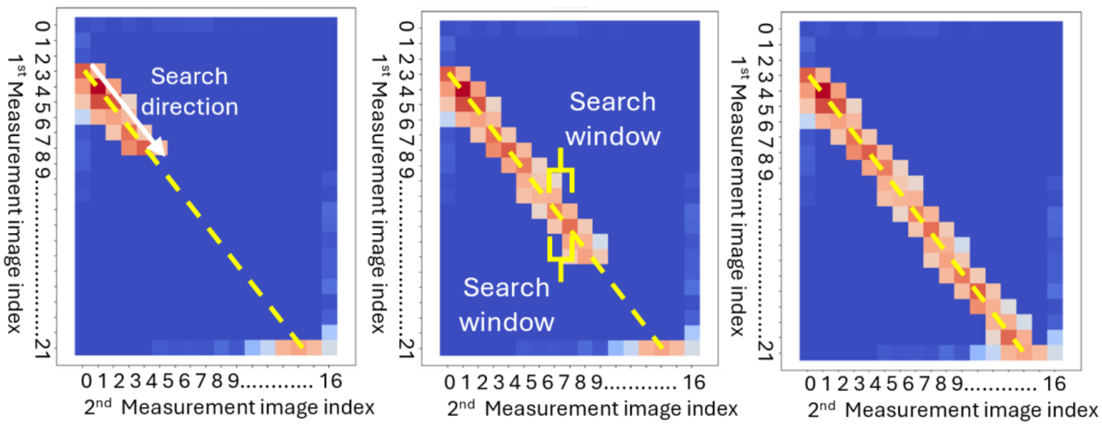


Figure 7 Searching along the line of best match with a small search window.

4. RESULTS AND CONCLUSIONS

After developing the search algorithm, it was implemented on several measurements to extract the best pairs for further analysis of chronological changes. An extracted example is shown in Figure 8.

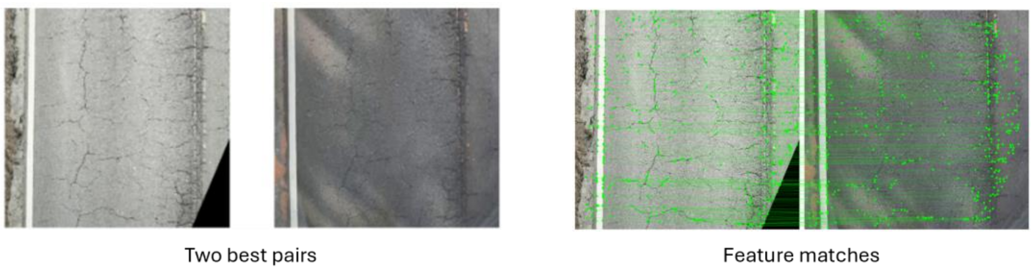


Figure 8 An example of matched pairs and their features.

In the preceding sections, the reduction in search time may not appear significant; however, as the number of images in the measurements increases, the time taken to search reduces substantially. Table 2 provides a comparison for an example.

Table 2 Time required by the different search techniques.

Measurement size	Basic (s)	Diagonal search* (s)	Line search** (s)	Time reduced (%) (Basic to line search)
12 x 9 images	31	25	18	41.9
22 x 26 images	163	69^	37	77.3
39 x 32 images	351	101^	62	82.3
79 x 89 images	1980	263	134	93.2

* Window search = 5 images
** Window search =1 image
^All image pairs not found

REFERENCES

Benedetto, A., & Pensa, S. (2007). Indirect diagnosis of pavement structural damages using surface GPR reflection techniques. *Journal of Applied Geophysics*, 62(2), 107-123. https://doi.org/10.1016/j.jappgeo.2006.09.001

Gao, W., Xue, K., Nagayama, T., Zhao, B., Su, D., & Xue, K. (2023). Rut depth estimation by distortion analysis of images taken by an in-vehicle camera. In *Life-Cycle of Structures and Infrastructure Systems* (1st ed., pp. 8). CRC Press. https://doi.org/10.1201/9781003323020

Kyriakou, C., Christodoulou, S. E., & Dimitriou, L. (2019). Smartphone-Based Pothole Detection Utilizing Artificial Neural Networks. *Journal of Infrastructure Systems*, 25(3), 04019019. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000489

Lindenberger, P., Sarlin, P.E., and Pollefeys, M. 2023. LightGlue: Local Feature Matching at Light Speed, *ICCV*.

Mehta, Y., & Roque, R. (2003). Evaluation of FWD Data for Determination of Layer Moduli of Pavements. *Journal of Materials in Civil Engineering*, 15(1), 25-31. https://doi.org/10.1061/(ASCE)0899-1561(2003)15:1(25)

Umashankar, B., Hariprasad, C., & Kumar, G. T. (2016). Compaction Quality Control of Pavement Layers Using LWD. *Journal of Materials in Civil Engineering*, 28(2), 04015111. https://doi.org/10.1061/(ASCE)MT.1943-5533.0001379

Xue, K., Nagayama, T., & Zhao, B. (2020). Road profile estimation and half-car model identification through the automated processing of smartphone data. *Mechanical Systems and Signal Processing*, 142, 106722. https://doi.org/10.1016/j.ymssp.2020.106722