

Original Research Paper

Nondestructive Test Using a 3D Computer Vision System for Damage Detection of Structures

Taylor M.V. Szeto and Sun Yi

Department of Mechanical Engineering,
North Carolina A&T State University, 1601 E. Market St., Greensboro, NC, 27411, USA

Article history

Received: 12-05-2018

Revised: 24-05-2018

Accepted: 01-6-2018

Corresponding Author

Sun Yi

Department of Mechanical
Engineering, North Carolina
A&T State University,
Greensboro, NC 27411, USA
Email: syi@ncat.edu

Abstract: Structures that do not have a definitive map nor a clearly known state of health may exist underground, beyond our reach and in unsuitable environments. Nuclear facilities contain underground tunnels that exhaust hazardous gases. Industry has miles of sewage lines beneath it that emit flammable and toxic gases. Regular inspection and maintenance is an essential part of failure prevention, however, these fatal environments have prohibited proper inspection of such infrastructure. Thus, the future of computer vision is vital to quality inspection. A strategically designed robot can be trained to visually inspect any structure, detect if there is a damage and decide if the damage is critical. Likened to any good inspector, the robot must be trained to investigate the nature of the damage and to alert the user of potential failures. This article discusses robotic training to detect damage in concrete structures and make decisions to the significance and impact of the defect.

Keywords: Nondestructive Testing, Computer Vision, Visual Inspection, 3D Camera, Concrete

Introduction

Visual inspection is the first step in analyzing the condition of a structure. Trained eyes are quick to recognize cracks, corrosion and potential failures in a structure without interfering with the subject. However, the human eye cannot operate in hazardous environments, nor can it store the data for further analysis. An automated robot can collect visual inspection data from the surface of a structure and analyze it, even with exposure to toxins. An efficient method for conducting a visual inspection in harmful environments utilizes computer vision systems in robotics. In this article, Intel's RealSense SR300 depth camera is used in tandem with MATLAB and RecFusion software to demonstrate nondestructive testing on structures. The depth camera measures the surface topology of concrete structures and the resulting data is analyzed for abnormalities in depth measurements. The results demonstrate the ability for a depth camera to detect specific damages in a structure and quantify them to make a decision.

It has become common for robots to inspect underground infrastructures due to hazardous conditions. Pipe inspection companies utilize lidar technology in

conjunction with robots to examine the degradation or buildup on concrete piping. Depth data can generate an up-to-date model of pipe, which can be compared with any initial or intended pipe parameters. Any significant difference in depth is an indication that the pipe has corroded or is prone to debris buildup. The applications for using depth cameras span from large-scale industrial inspection to everyday quality maintenance. In the following experiments, the RealSense camera generated 3D models of concrete structures, which were compared with the intended parameters of the structures to detect damage in the form of surface cracks.

Related Work

The performance of Intel SR300 depth camera as a 3D scanner was characterized by Carfagni *et al.* (2017). The device's performance is assessed by applying the VDI/VDE Normative, which is the international standard for characterization of measuring devices. The raw data (unfiltered) is analyzed in order to account for the worst possible conditions a user might experience. The author concludes that the SR300 outperforms most competitor cameras when used as a close-range 3D scanner.

The SR300 is used for fast, accurate small-scale 3D scene capture in the works of Carey *et al.* (2017) by adding lenses and precise calibration, the device is used for tracking termite activity at high speeds with mm-level accuracy. The works demonstrates high accuracy and high-speed capabilities of the SR300 depth camera.

Unsupervised hot spot detection using the SR300 was introduced by Chen *et al.* (2017) touch patterns are acquired from contour searching in depth. The work uses the depth camera to capture touch and hand motion and detect hotspots and interactions within the user experience. The applications for this works can cover continuous data collection of external forces that come in contact with infrastructure.

Augmented reality maintenance training was introduced using the SR300 depth camera in Abdelnaby *et al.* (2017) the works focused on interactivity between the user and virtual object using hand gestures to manipulate virtual scene objects. The purpose is aimed to stabilize gesture recognition and picking selection.

Gong *et al.* (2017) proposed an algorithm for point cloud segmentation sampled by the SR300 depth camera. Segmentation of objects aids in bin-picking applications for robotics. The ability to guide a robot to an object within a bundle of objects has applications that span from small-scale bin picking to large-scale inspection.

Materials

The experiments were designed with structural inspection as a priority target. Failure can be detected in structures with cracks, corrosion, buildup of debris and discoloration. A device was needed that detected a defined jump in depth from one pixel to the next and had the ability to display an RGB video stream.

Experiments were conducted with the Intel RealSense SR300 depth camera and two software programs, MATLAB and RecFusion. The experiments were split into two phases. Phase one utilizes MATLAB in order to scan in a translational plane and return depth frames of the surface. Phase two utilizes RecFusion to rescan the damage point in a rotational frame in order to generate a 3D model of the damage.



Fig. 1: Intel RealSense RS300 depth camera

The RealSense SR300 depth camera, shown in Fig. 1, was used to conduct the experiments presented here. The camera was chosen for its compact size and ability to implement short-range 3D imaging. The device synchronizes depth, color and infrared video, which allows texture mapping of depth to color. The infrared camera captures depth from 0.2 to 1.5 m and has a depth resolution of 640×480 Video Graphics Array (VGA).

Phase one of each experiment utilized the Image Acquisition Toolbox in MATLAB for data management. Scans were taken in a translational motion across the entirety of the surface being imaged. While scanning, the camera was positioned one meter from the original surface. Two video feeds were transmitted into the MATLAB workspace. The user has the option of searching the colored (RGB) video feed for observed damage or searching the depth stream for critical damage with specific margins.

During the second phase of each experiment, scans were taken in a rotational motion about the center axis of the damage. RecFusion software was used to generate a 3D reconstruction of the critical damage. The image can then be measured to check the accuracy of phase one data and verify if the damage is indeed critical.

Results

Undamaged Concrete

Light reflections and shadows can be misconstrued as damage when imaging with a depth camera. In order to combat the noise in the following images, a control concrete slab was scanned for an average surface depth, shown in Fig. 2. The image is a mesh plot from MATLAB. If it were to be rotated, a 3D model of the concrete surface can be seen, this will be shown later in the damaged concrete models. The horizontal X-axis shows the distance along the surface of the concrete slab, which is measured in pixels. The resolution of the depth camera allows for a 640 pixel scan in the X-direction and each pixel has a corresponding depth value. The vertical Y-axis shows the depth of the concrete slab, which is measured in micrometers. The yellow colored portion of the figure shows noise, where the blue colored portion shows the roughness of the surface of the concrete slab. The control slab was useful in analyzing the damaged concrete because of the ability to ignore smaller deviations from the average surface depth. The control concrete was measured in MATLAB using the Image Acquisition Toolbox. A depth image was captured and imported to the MATLAB workspace. A matrix is returned in the workspace with a depth value for each pixel displayed. A code was generated to diminish the light pollution by simultaneously imaging with RGB and depth. With the information in Fig. 2, an average surface depth was observed to be at 6×10^4 when imaged from 18 inches distance.

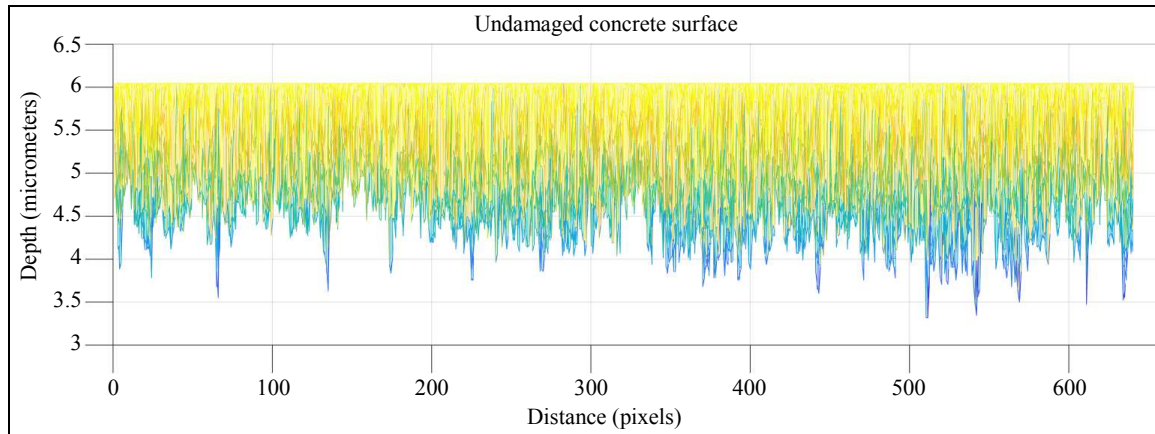


Fig. 2: Depth of Flat concrete surface in MATLAB

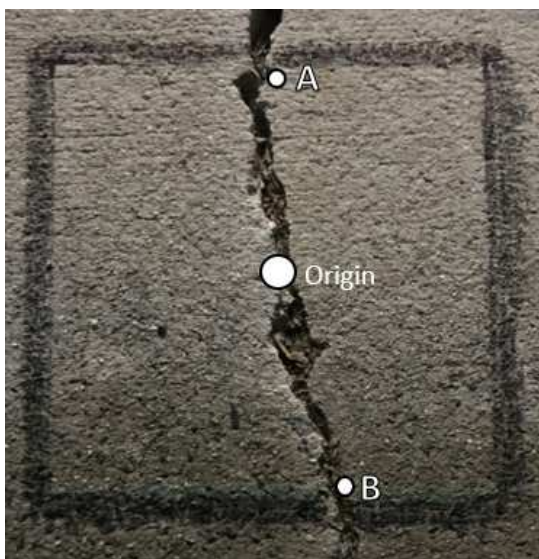


Fig. 3: Original Concrete Floor Image Pre-processing

The following experiments use a deviation from the intended surface depth to detect damage. The first experiment examines concrete with a damage that exhibits cracking, where the depth is greater than the average surface depth. The second experiment examines concrete with a damage point that exhibits debris build up, where the depth is less than the average surface depth.

Damaged Concrete

Next to the control concrete subject, a damaged concrete subject was observed with a noticeable crack and shifting, shown in Fig. 3. The right side of the concrete was observed to be sunken into the ground, while the left side appeared to be at its intended position. The crack was easily observed from the colored image, but the shifting in the concrete slabs was not intuitive. The depth of shifting, depth of crack and the width of crack were measured with dial calipers and recorded in Table 1.

Table 1: Measurements of Concrete Crack [values in inches]

	Shift depth	Crack depth	Crack width
Point A	0.365	1.245	0.339
Point B	0.410	0.810	0.333
Origin	0.417	0.951	0.413
Average	0.397	1.002	0.362

Three shifting depth measurements, three crack depth measurements and three width measurements were taken of the damaged concrete subject, at points A, B and the origin, in order to obtain an average depth and width measurement of the crack. Measurements for the crack depth were measured from the top of the debris inside the crack to the left side of the concrete (original position).

Phase I: Analysis Using MATLAB

The damaged concrete was imaged using MATLAB's Image Acquisition Toolbox and was imported into the MATLAB workspace. The returned data took the form of a matrix with 640x480 dimensions with each pixel containing a depth value. Any depth value that was greater than the average surface depth from the control concrete was trimmed. Trimming out extreme depth values ensured minimal light reflection to interfere with the true data. The "mesh" tool in MATLAB plots was used to generate a wireframe that interconnects the depth values per pixel to portray a 3D model of the subject, shown in Fig. 4. The image was taken at an 18 inch distance from the concrete subject. In this mesh graph, the user can manipulate the view to measure depths at specific pixel locations. In this experiment, the average is being compared, so there is not a need to examine every row and column of pixels. However, a specific data set can be selected by locating the desired pixel and measuring specific dimensions.

In MATLAB's Image Acquisition toolbox, the depth value returned is in micrometers and the pixel values are in pixels. In order to analyze the accuracy of the depth

measurements, the measured values in MATLAB were compared with the actual values. For depth measurement, the MATLAB mesh plot of the depth values was rotated to view the spike in depth where the crack is located, shown in Fig. 5. The measurements were converted to US customary units for comparison, Equation 1-1. A summary of measurements is located in Table 2.

$$6 * 10^4 - 2 * 10^4 = 4 * 10^4 \mu m = 1.57 in \quad (1-1)$$

Table 2: Measurements of concrete crack in MATLAB [values in inches]

	Shift Depth	Crack depth	Crack Width
Actual	0.397	1.002	0.362
MATLAB	unclear	1.57	unclear
Percent Error	n/a	56.7%	n/a

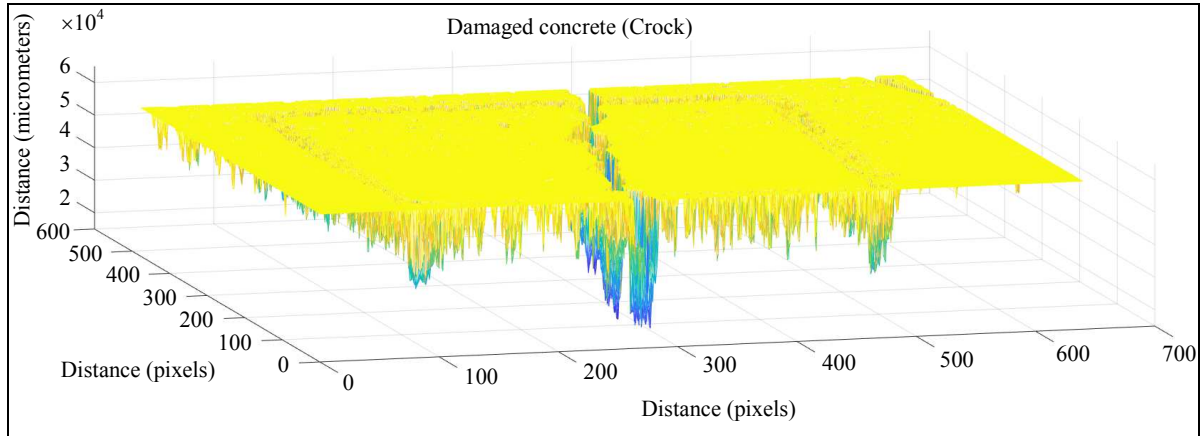


Fig. 4: Depth of surface crack in MATLAB

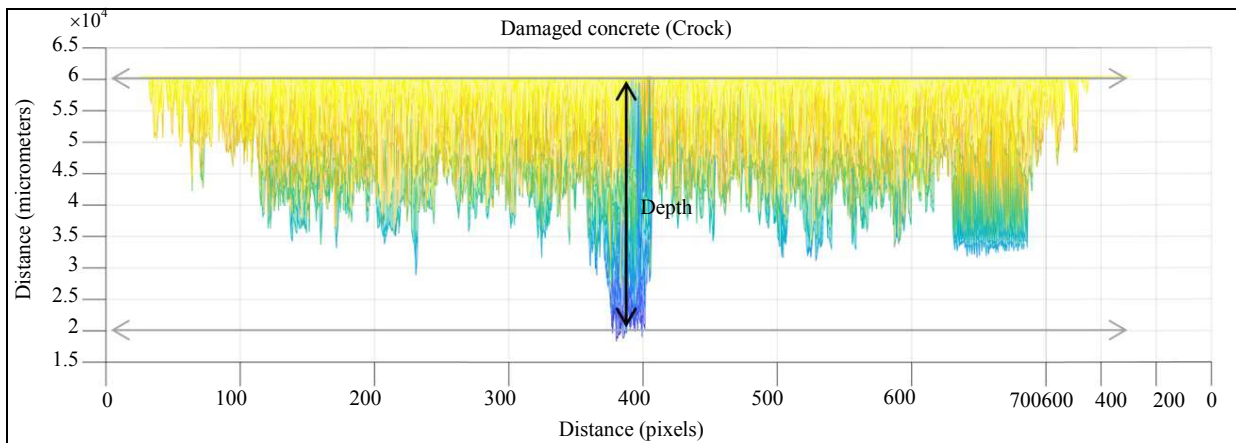


Fig. 5: Rotated View of Depth of Surface Crack in MATLAB

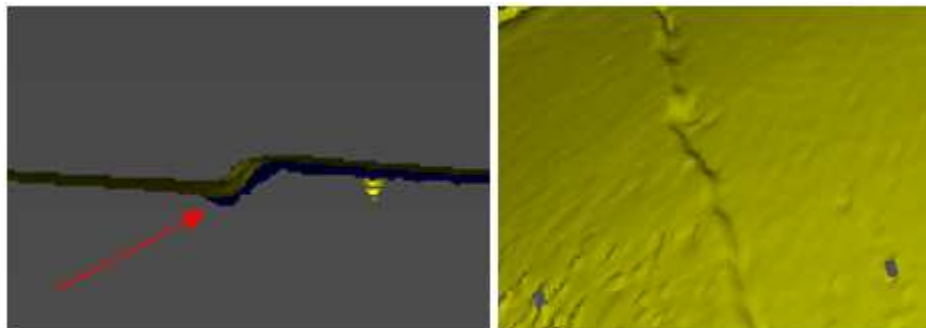


Fig. 6: (left) Concrete Floor Model in RecFusion, (right) Rotated View of Concrete Floor in RecFusion

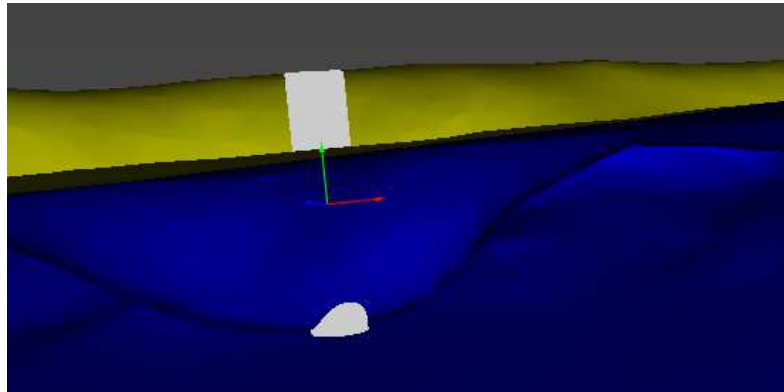


Fig. 7: Rotated view of depth of surface crack in MATLAB

Table 3: Measurements of Concrete Crack in RecFusion [values in inches]

	Shift depth	Crack depth	Crack width
Actual	0.397	1.002	0.362
RecFusion	0.315	0.906	0.394
Percent Error	20.7%	9.6%	8.84%

Phase 2: Analysis using RecFusion

The damaged concrete was imaged at a one meter distance from the concrete subject with RecFusion. A center axis for the concrete floor was determined using the linear endpoints at points A and B in Fig. 2. The concrete was imaged by slow-scanning along a 90° arc, while maintain a distance of one meter. Using the post processing techniques, a model of the concrete slab can be seen in Fig. 6.

RecFusion allows the user to see more detailed information about the depth data. In Fig. 6, one can observe shifting in the concrete in addition to the crack depth. RecFusion socket tool was used in post processing to analyze scaling accuracy, shown in Fig. 7. Using the socket tool, it was determined that the average depth and width of the crack were as follows. A summary of the measurements taken from RecFusion are recorded in Table 3.

RecFusion returned an accurate depth measurement of the concrete damage. Also, RecFusion allowed observation of shifting in concrete, whereas MATLAB data is hard to distinguish any shifting.

Discussion

It is advantageous to scan a concrete surface for discontinuities and assess the discontinuities for severity. The use of MATLAB can alert the user of an abnormality in depth and a rescan of the damage with RecFusion can account for accurate measurements of the damage. MATLAB can scan large data and store this data for further analysis. A code may also be generated in MATLAB to ignore the smaller jumps in depth that

may be due to surface roughness. This can be done by first scanning an undamaged concrete and observing a typical range that the material deviates from the average surface depth. This way only specific damages will be detected and smaller cracks can be eliminated from the data set. However, all data will be stored in case the next maintenance inspection shows the smaller crack has grown, the user can then look back and analyze the rate at which the damage has progressed.

Using the SR300 depth camera, MATLAB returns pixel data which is difficult to interpret measurements. Due to the limitations of MATLAB, the software was used in conjunction with RecFusion. RecFusion can scan small data and create a 3D model with accurate measurements. Combining the capabilities of both softwares allows the user to meet the requirements of finding and accurately representing damage. Knowing the location of a damage and an accurate representation of what the damage looks like, a code can be generated to assess if the damage is severe to the infrastructure.

The detection of damage is dependent on the user's definition of damage. In the experiment conducted above, MATLAB returned a frame with depth data that was outside of the controlled depth data from the undamaged surface. The frame can be assessed by searching the array for values that exceed the limits of the controlled depth data. For instance, in Fig. 2, the limits of the data are about $3.75 \times 10^4 \mu\text{m}$ and $6 \times 10^4 \mu\text{m}$. When depth data is below or above these limits, it is an indication that there is a discontinuity in the structure. Once a discontinuity is returned in the MATLAB workspace, the area can be rescanned to generate a 3D model in the RecFusion workspace, then the damage can be assessed for severity.

Conclusion and Future Works

A computer vision system using the SR300 depth camera is capable of detecting damage in concrete. With a section of undamaged structure being the controlled

group, depth data of damaged structures can be compared with the intended parameters of the design. This work proves that a depth camera can be used in conjunction with commercially available software to assess damages to concrete structures.

A damage detection system should be fully capable of detecting cracks, debris build up, discoloration and any other abnormality of a structure. Future works will include the assessment of debris build up and any addition of material on a structure that could lead to failure. Debris buildup is an important detection in concrete piping, where buildup could cause stagnation of flow and rupture of piping infrastructure.

Funding Information

This work was supported by the Department of Energy Minority Serving Institution Partnership Program (MSIPP) managed by the Savannah River National Laboratory under SRNS contract DE-AC09-08SR22470.

Author Contributions

Taylor Szeto: Is the lead author of this work. She was responsible for researching related works, conducting experiments and analyzing data.

Sun Yi: Acted as an advisor of this work. He was responsible for ensuring the reproducibility of the experiments as well as helping revise the article.

Ethics

This article is an original research paper. There are no ethical issues that may arise after the publication of this manuscript.

References

- Carfagni, M., R. Furferi, L. Governi, M. Servi and F. Uccheddu *et al.*, 2017. On the performance of the Intel SR300 depth camera: Metrological and critical characterization. *IEEE Sensors J.*, 17: 4508-4519. DOI: 10.1109/JSEN.2017.2703829
- Carey, N., J. Werfel and R. Nagpal, 2017. Fast, accurate, small-scale 3D scene capture using a low-cost depth sensor. *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, Mar. 24-31, IEEE Xplore Press, Santa Rosa, CA, USA, pp: 1268-1276. DOI: 10.1109/WACV.2017.146
- Abdelnaby, M., M.A. Elazem, H.A. Aly and A. Kaboudan, 2017. Augmented reality maintenance training with intel depth camera. *Proceedings of the International Conference on Machine Vision and Information Technology*, Feb. 17-19, IEEE Xplore Press, Singapore, pp: 116-122. DOI: 10.1109/CMVIT.2017.9
- Chen, L., K. Kondo, Y. Nakamura, D. Damen and W.W. Mayol-Cuevas, 2017. Hotspots detection for machine operation in egocentric vision. *Proceedings of the 15th IAPR International Conference on Machine Vision Applications*, May 8-12, IEEE Xplore Press, Nagoya, Japan, pp: 223-226. DOI: 10.23919/MVA.2017.7986841
- Gong, X., M. Chen and X. Yang, 2017. Point cloud segmentation of 3D scattered parts sampled by RealSense. *Proceedings of the IEEE International Conference on Information and Automation*, Jul. 18-20, IEEE Xplore Press, Macau, China, pp: 1-6. DOI: 10.1109/ICInfA.2017.8078881