Realtime damage detection in long conveyor belts using deep learning approach

Uttam Kumar Dwivedi [ab], Ashutosh Kumar [b] and Yoshihide Sekimoto [b].

[a] Hazama Ando Corporation, Japan

[b] Department of Civil Engineering, University of Tokyo, Japan

E-mail: uttam.kumar.dwivedi@ad-hzm.co.jp, uttamdwivedi04@gmail.com, sekimoto@iis.u-tokyo.ac.jp

Abstract —

Japan's Road networks rely heavily on the mountain tunnels due to its topology. During the construction of these tunnels, mucking process is conducted to remove crushed rocks and rubbles in the tunnel with the use of long conveyor belts with the size of 3 to 10km. Regular visual inspections of these belts are carried out tediously by the workers to ensure belt integrity. To reduce the burden on workers, the paper proposes a vision based deep learning solution deployed on an edge device. Proposed framework detects the size of damage ranging from 1 cm to 100cm. Edge device deployment helps the workers to receive the result in real-time regardless of internet availability or working conditions. The effectiveness of the proposed framework is confirmed on 3 tunnel construction sites, with the estimated mean average precision of 85% for crack detection. The study can be applied in other domains of construction industry such as road damage or concrete damage categorization.

Keywords —

Internet of things (IoT); Edge computing; Deep learning; Smart construction; Damage detection; Real-time

1. Introduction

Japan has a long-standing reputation for excellence in construction engineering, particularly in the area of tunnels. This is due to the country's complex geology and densely populated urban areas. Japan has a vast network of road and railway tunnels, with recent data showing that there are over 5000 kilometers of road tunnels and 3800 kilometers of railway tunnels [1][2]. With the advancement of new tunneling technologies [3][4], there is renewed focus on sustainable development through shorter construction periods, cost savings, environmental preservation, and improved quality. One of the frequently used methods is the New Austrian Tunneling Method (NATM) [5][6], which is the basis of modern tunneling techniques. This method involves breaking rocks using explosives, followed by removal of the crushed rocks using conveyor belts. However, sharp rocks of various sizes can damage the conveyor belts, leading to accidents or belt failures [7]. It is therefore essential to regularly inspect the belt surfaces for damage to prevent such occurrences.

The inspection of conveyor belts in mountain tunnel construction continues to rely on manual visualization, which is a strain on inspectors and reduces the frequency of inspections due to fatigue. Additionally, stopping the conveyor belt during inspections decreases the work efficiency of the mucking process and causes delays. To address these problems, a new edge AI based deep learning framework is proposed to automatically detect damage in real-time and provide real-time alerts to safety engineers. The proposed system consists of three parts: damage detection, frame tracking, and real-time alerting. The effectiveness of the proposed method has been tested on three different mountain tunnel construction sites and has shown potential to enhance productivity and safety with an average mean average precision of 0.85 for damage detection. This solution uses deep learning and image processing on an offline edge device [8] to provide a real-time alerting platform for damages on long conveyor belts.

2. Literature Review

Automated inspection of tunnel constructions is a challenging but fascinating area for researchers in the construction field [9][10][11]. Over time, various devices have been developed to address these issues. Infrared thermal imaging technology and X-ray based nondestructive techniques [12] have been devised to detect damage in coal mines. These methods can detect cracks, but they require specific image processing and high-end, expensive equipment. Recently, AI-based methods, especially deep learning algorithms [13][14], have been widely used in image classification, detection, and segmentation. These methods use high-speed CMOS cameras and high-performance computing devices to extract features from images. Researchers have proposed MATLAB's deep learning solution[15] with two-layer neural networks to detect and locate conveyor belt damage in real-time. These methods provide a good balance of network architecture depth, image resolution, detection speed, and mean average precision (mAP). They can locate damages of different sizes and identify the numbers marked on the side of conveyor belts, which indicate the distance and make repair easier.

3. Research methodology

The methodology for this study has been framed around three main areas: data collection to include a range of scenarios, experimental setup and applied computer vision techniques-

A. Data collection

The conveyor belt crack detection (CBCD) dataset was developed by collecting 9,362 images from various mountain tunnel construction sites. The images were passed through data annotation tool [16] to create bounding box around different cracks. Figure 1(A) shows a sample of data, where the damage area is annotated with class id. The annotations are saved for each image as the equation [1].

$$< class\ id > < \frac{Xo}{X} > < \frac{Yo}{Y} > < \frac{W}{X} > < \frac{H}{Y} > \quad (Eq. 1)$$

[Where, class id = label index of the class to be annotated; Xo = X coordinate of the bounding box's center.

Yo = Y coordinate of the bounding box's center.

W = Width of the bounding box.

H = Height of the bounding box.

X = Width of the image

Y = Height of the image]

The CBCD dataset is split into training, validation and testing set with 70%, 20% and 10% respectively.

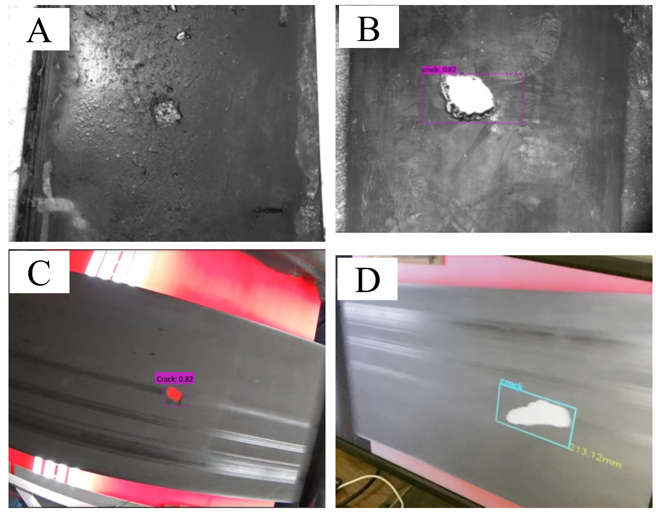

Figure 1. Training dataset for Conveyor belt crack detection (CBCD). (A) shows reflecting surface with non-through cracks, (B) shows through cracks while (C) shows through cracks with orange lights reflecting through it. (D) show scratched surface detected by AI. B and C also shows the bounding box around the crack area.

B. Experimental setup

Figure 2 shows a setup diagram is shown. The camera was positioned at the ground facing upwards with 1.5 meters between it and the conveyor belt. This configuration was intended to be used in construction scenarios. In order to maintain the belt's cleanliness, water from a stream was poured onto the belt to wash away any sled debris. A dark room was prepared with a camera mounted inside and lighting set up next to it. This provided optimal lighting conditions for filming.

Cameras require a specific lighting condition to properly record footage. A second LED light with an orange hue was added to the camera setup to indicate the severity of damage. Camera equipment used at construction sites in mountain tunnels has a rubber, polyester, and rubber layer construction. Each layer is 10mm thick and the total width of the belts is 0.6 meters. The conveyor belts operate at average speed of 200m/min.

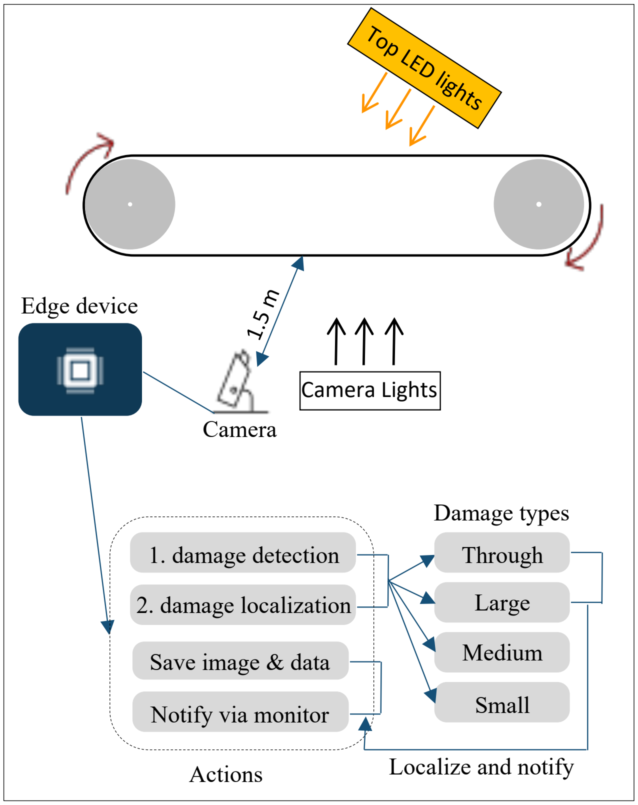

Figure 2. Simplified diagram of experiment setup and expected output. Edge device's deep learning and image processing program provides output. Top LED orange lights provide contrast with respect to bottom white lights.

Table 1. Hyperparameters of the YOLOv5 network for training on the CBCD dataset.

Hyperparameter	**Value**
Input size	608
Learning rate	0.001
Batch size	64
Sub-division	16
Optimizer	SGD with momentum

C. Applied computer vision techniques

The section provides the details of deep learning models and image processing techniques used in proposed paper.

a) Deep learning-based detector model –

Object detection is used for identifying the location of cracks. Proposed paper uses a single instance target detection algorithm YOLOv5 [17] as a deep learning model to detect cracks in real-time. Since the original model is trained on the COCO dataset [18] with 80 general classes like person, cars, trucks etc. However, the original model doesn't include specific object type such as cracks in conveyor belts.

In general, the training dataset contains around 200,000 images to train the model from scratch. With the help of transfer learning technique [19], where the original pretrained trained model is reused as a starting point for a model on second task. We train YOLO model using around 6,550 images from the CBCD dataset for 30,000 iterations with a batch size of 64 using the initial pre-trained weights from ImageNet dataset [20] for the first 137 convolutional layers. For training, NVIDIA RTX 3090 with 24GB memory was used for continuous 10 hours. The hyperparameters for the training is shown in Table 1.

b) Optimization of the neural network –

To run the detection framework effectively in real-time on the edge devices, proposed paper optimizes the YOLOv5 model. Neural networks generally use FP32 (32-bits floating point precision) [21] to store parameters such as weights and biases. Using a higher precision increases computational complexity and increases the size of the model. Through experiments, it has been found that the neural network model using half-precision FP16 (16-bit floating point) as parameters has similar performance to the neural network model using single-precision FP32. Therefore, the precision can be reduced to FP16 without severe loss of performance. This may be because neural networks are very resilient to noise. Decreasing the precision value from FP32 to FP16 is considered to introduce noise. Furthermore, half-precision models are very lightweight and show a significant increase in inference speed compared to single-precision models [21]. Proposed paper performs optimizations in the TensorRT framework [22] by reducing floating-point precision to FP16 and incorporating layers that perform conventional operations.

c) Crack size estimation –

Proposed paper deploys the optimized TensorRT model on the edge device NVIDIA Jetson NX, which localizes the cracks. To estimate the crack size in metric units, a distance estimation technique proposed by

Karney et al. [23] is used. Their approach is essentially limited to estimating the distance of an object when its true size is known and the focal length, camera sensor size and image resolution are fixed. In our approach, the distance from the camera to the conveyor belt is fixed (1.5 meter) as shown in Figure 2. The only parameter that remains to be determined is the crack size, which can be evaluated using Equation 2.

$$\mathcal{H}_{crack} = \frac{\mathrm{d} \times h_{crack,px} \times \mu_h \times 1000}{f \times \mathrm{I}_h} \quad (Eq.2)$$

[Where, $\mathcal{H}_{crack}$ = Crack size in metric units (mm); d = Fixed distance of the conveyor belt surface from the camera; $h_{crack,px}$= Height of the crack in pixels obtained from the bounding box; μ_h = Height of the camera sensor; f = Focal length of the camera; I_h =Height of the image resolution]

Size of the cracks are divided into 3 range as large damage size (H_{crack} >10cm), medium damage (10cm< H_{crack} <5cm) and small damage (H_{crack} <5cm) shown in the table 2.

4. Result

Three different mountain tunnel construction sites were selected to run the crack detection and size estimation framework on long conveyor belts. The trained YOLOv5 model on CBCD dataset achieved a combined mean average precision (mAP) of 87% with FP32 and 85% with optimized FP16-TRT [21] on small edge device. The test was run on 500 images completely different from original CBCD dataset. However, the optimized model with TensorRT framework (FP16-TRT) provides significance speed on the edge device NVIDIA Jetson NX [8] from 5 FPS (frames per second) to 15 FPS as shown in the Figure 3. The frames per second (FPS) is calculated by averaging the inference FPS for 5,000 iterations. Figure 1 (D) shows a sample of the detection result with colored bounding box around the crack area with estimated size of 213mm (21.3 cm).

In Table 2, proposed paper shows the accuracy of crack detection by its size. The results presented in Table 2 are based on crack detection results carried out at the mountain tunnel site using NVIDIA Jetson Xavier NX device. We collect the samples from the image frames of the moving conveyor belt. Thus, the light reflections, water and dust on the conveyor belts require us to capture the testing data using multiple cameras at different locations and angles. We notice that a very small false positive for no damages, while the accuracy of crack detection reduces as the size of the crack reduces. We achieve the highest accuracy of 89.23% for large damages and the lowest accuracy of 64.13 for smaller damages as smaller damages can get difficult to detect due to light reflections or dust appearing on top of conveyor belt.

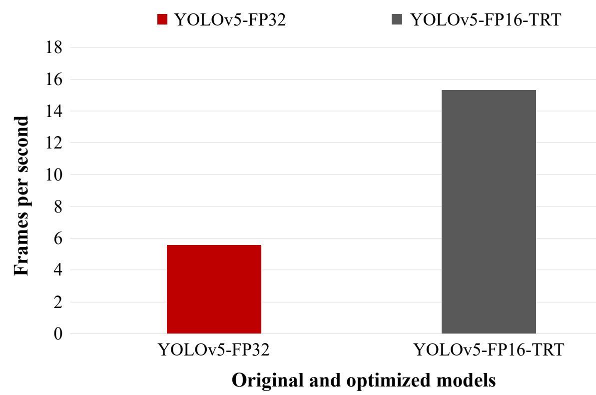

Figure 3. The comparison of inference speed of the original YOLOv5 model and its optimized version. Comparison of YOLOV5(YOLOv5-FP32) with 608x608 input resolution and its optimized version in TensorRT (YOLOv5-FP16-TRT). The frames per second (fps) is calculated by averaging inference fps for 5,000 iterations.

Table 2: Table shows the accuracy of the detection for various crack sizes

Damage size	# Of sample	Detected	Not detected	Accuracy (%)
Large damage (size >10cm)	103	92	11	89.3
Medium damage (10<size <5cm)	120	91	29	75.8
Small damage (size <5cm)	92	59	33	64.1
Through damage	34	28	6	82.4
No damage	500	4	496	99.2

5. Discussions and Conclusions

Long conveyor belts are widely used in various industries for transporting rocks, sand, and other heterogeneous materials over long distances. However, the manual inspection of these belts to detect damage is time-consuming and expensive, and current methods that use infrared laser light, x-rays, sound, magnetic, and ultrasound energy are also costly and have limited ability to detect large damage. In this study, we propose a simple and cost-effective method to detect and locate conveyor belt cracks in real-time. Our proposed system runs on edge devices and uses a server that operates offline. The system uses a CBCD dataset of 9,362 images to train a YOLOv5 model, optimized to execute inference on lightweight devices. The optimized model can detect cracks and track on the conveyor belt with a mean average precision of 85% and can process 15 frames per second on resource-constrained devices. The severity of the damage is estimated by fixing the camera's distance from the conveyor belt and using a monocular camera to categorize the size of the damage. Further improvement can be made by combining multiple video frames to identify crack regions in the conveyor belt.

At this stage, our proposed model for detecting and locating conveyor belt cracks in real-time is novel and compares favorably to existing methods in the field. To check the novelty of our proposed model, we reviewed the literature for belt tear detection, a vision-based method developed by Guo et al. [13] detects large size damages using YOLOv5-m [17] with a mean average precision (mAP) of 82.5%. However, this method fails to identify small or medium size damages. Similarly, Agata et al. [15] proposed an artificial intelligence-based approach for the classification of conveyor belt damage using a two-layer neural network and achieved an accuracy of 80%. Another method based on Haar-Ada Boost and Cascade algorithm was proposed by Wang et al. [14], where longitudinal tears of a conveyor belt under uneven light were detected with an accuracy of 97%. These methods can only detect large types of damage, whereas our proposed method can detect various types of damage and improve the overall detection accuracy, as shown in Table 2.

In conclusion, our proposed method provides a simple, inexpensive, and sustainable solution for detecting and locating conveyor belt cracks in real-time. The optimized model has a high accuracy and can detect both cracks and digits on the conveyor belt, while categorizing the severity of the damage using a monocular camera. Our proposed method has the potential to revolutionize the field and make manual inspection of conveyor belts a thing of the past. A more detailed study will be presented in an upcoming journal.

6. Acknowledgement

This study is supported by Tokyo Kizai Kogyo co. ltd. and University of Tokyo, Japan. The authors declare no conflict of interest. We acknowledge Mr. Takeshi Hosokawa of Tokyo Kizai Kogyo Co., Ltd., and Mr. Tsuneo Koike of Ando Hazama Corporation for their cooperation in filming the continuous belt conveyor of this research.

7. Conflict of Interest

There are no conflicts to declare.

8. Reference

1. [Internet]. Japan-tunnel.org. 2022 [cited 12 July 2022]. Available from: https://www.japan-tunnel.org/en/sites/www.japan-tunnel.org.en/files/tnnl_book_aspects/Tunnel%20Activity%202020%20Overview_0.pdf
2. [Internet]. Ejrcf.or.jp. 2022 [cited 12 July 2022]. Available from: https://www.ejrcf.or.jp/jrtr/jrtr66/pdf/38-51.pdf
3. Miura K. Design and construction of mountain tunnels in Japan. Tunnelling and Underground Space Technology. 2003 Apr 1;18(2-3):115-26.
4. Karakuş M, Fowell RJ. An insight into the new Austrian tunnelling method (NATM). Proc. ROCKMEC. 2004 Oct 21.
5. Tran TH. A Study on Tunnel Lining Concrete with Crushed Aggregate from NATM Muck. InProceedings of the 3rd International Conference on Sustainability in Civil Engineering 2021 (pp. 145-151). Springer, Singapore.
6. Phadke V, Titirmare N. Construction of tunnels, by new austrian tunneling method (NATM) and by tunnel boring machine (TBM). International Journal of Civil Engineering (IJCE). 2017 Oct 21;6(6):25-36.
7. Bortnowski P, Kawalec W, Król R, Ozdoba M. Types and causes of damage to the conveyor belt-review, classification and mutual relations. Engineering Failure Analysis. 2022 Jun 16:106520.
8. Feng H, Mu G, Zhong S, Zhang P, Yuan T. Benchmark analysis of Yolo performance on edge

intelligence devices. Cryptography. 2022 Apr 1;6(2):16.
9. Montero R, Victores JG, Martinez S, Jardón A, Balaguer C. Past, present and future of robotic tunnel inspection. Automation in Construction. 2015 Nov 1;59:99-112.
10. Gambao E, Balaguer C. Robotics and automation in construction [Guest Editors]. IEEE Robotics & Automation Magazine. 2002 Mar;9(1):4-6.
11. Trybała P, Blachowski J, Błażej R, Zimroz R. Damage detection based on 3d point cloud data processing from laser scanning of conveyor belt surface. Remote Sensing. 2020 Dec 25;13(1):55.
12. Ming-sheng W, Zheng-shi C. Researching on the linear X-ray detector application of in the field of steel-core belt conveyor inspection system. In2011 International Conference on Electric Information and Control Engineering 2011 Apr 15 (pp. 701-704). IEEE.
13. Guo X, Liu X, Zhou H, Stanislawski R, Królczyk G, Li Z. Belt Tear Detection for Coal Mining Conveyors. Micromachines. 2022 Mar 17;13(3):449.
14. Wang G, Zhang L, Sun H, Zhu C. Longitudinal tear detection of conveyor belt under uneven light based on Haar-AdaBoost and Cascade algorithm. Measurement. 2021 Jan 15;168:108341.
15. Kirjanów-Błażej A, Rzeszowska A. Conveyor Belt Damage Detection with the Use of a Two-Layer Neural Network. Applied Sciences. 2021 Jun 13;11(12):5480.
16. [Internet]. Tzutalin (2015), 'LabelImg', Free software: MIT License. Available from: https://github.com/heartexlabs/labelImg
17. Nelson J. Yolov5 is here [Internet]. Roboflow Blog. Roboflow Blog; 2021 [cited 2022Nov24]. Available from: https://blog.roboflow.com/yolov5-is-here/
18. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft coco: Common objects in context. InEuropean conference on computer vision 2014 Sep 6 (pp. 740-755). Springer, Cham.
19. Torrey L, Shavlik J. Transfer learning. InHandbook of research on machine learning applications and trends: algorithms, methods, and techniques 2010 (pp. 242-264). IGI global.
20. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. Imagenet: A large-scale hierarchical image database. In2009 IEEE conference on computer vision and pattern recognition 2009 Jun 20 (pp. 248-255). Ieee.
21. FP32 (Floating point format for Deep Learning) [Internet]. OpenGenus IQ: Computing Expertise & Legacy. 2022 [cited 12 July 2022]. Available from: https://iq.opengenus.org/fp32-in-ml/
22. Jeong E, Kim J, Ha S. TensorRT-based Framework and Optimization Methodology for Deep Learning Inference on Jetson Boards. ACM Transactions on Embedded Computing Systems (TECS). 2022.
23. Karney CF. Transverse Mercator with an accuracy of a few nanometers. Journal of Geodesy. 2011 Aug;85(8):475-85.