JCSE Journal of Computing Science and Engineering

Regular Paper
Journal of Computing Science and Engineering,
Vol. 19, No. 2, June 2025, pp. 61-71

Lightweight Crack Learning Model Using Morphology for Infrared Sensor Images

Joonho Byun
Department of Artificial Intelligence, Sogang University, Seoul, Korea
cdcd1115@gmail.com

Jeong-Joon Kim and Siwoo Byun*
Department of Software, Anyang University, Anyang, Korea
jjkim@anyang.ac.kr, swbyun@anyang.ac.kr

Abstract

Since various cracks can lead to significant loss of life and property, early detection and repair of cracks in structures is a very important procedure to prevent social losses. However, since it is difficult to detect cracks under paint or wallpaper using conventional techniques, it is necessary to use infrared-based thermal devices. In order to improve the performance of crack detection using infrared thermal imaging, we analyzed various image preprocessing methods and deep learning models. We developed a lightweight deep learning model using morphology preprocessing and found that morphology can be used to effectively highlight the boundaries and shape of the cracks. In particular, the lightweight Yolo model used with morphology had a faster processing speed and a similar level of accuracy with a smaller number of parameters. We demonstrate that lightweight deep learning can perform similarly to heavier models when appropriate image preprocessing is applied, which is useful in low-end portable devices.

Category: Computer Vision and Graphics

Keywords: Lightweight deep learning; Crack detection; Morphology; Thermography; Preprocessing

I. INTRODUCTION

Various cracks in structures (Fig. 1), if not properly repaired, can cause not only aesthetic problems but also a decrease in the strength of the structure, which can lead to significant loss of life and property. Therefore, early detection and repair of cracks in structures is a very basic and important procedure to prevent social losses [1-4].

In particular, the number of defect claims is increasing daily due to the proliferation of apartment buildings. A large part of the increase in defect lawsuits is due to the judgment of excessive defect repair costs compared to the actual defect repair costs. Among them, the cost of repairing cracks accounts for a large proportion.

Concrete crack investigation and inspection methods typically performed on concrete structures are spot checking. This method is subject to the subjectivity of the inspector, which can reduce the objectivity and reliability

Open Access http://dx.doi.org/10.5626/JCSE.2025.19.2.61 http://jcse.kiise.org

Received 13 August 2024; **Accepted** 24 September 2025
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

Fig. 1. Example of a crack [11].

of the record, and it is difficult to determine the progression of damage if the inspector changes. Therefore, in order to overcome these problems and improve the objectivity and accuracy of damage inspection and the convenience of data recording and storage, an image processing method is needed that automatically extracts the results of visual inspection by capturing an image through an imaging device [5].

However, it is difficult to detect cracks under paint or wallpaper using conventional image processing techniques, so we aim to overcome this problem by using infrared-based thermal images and lightweight deep learning models for portability. In other words, this paper proposes a crack detection model using infrared images and lightweight deep learning, and verifies its effectiveness through experiments.

In practice, thermal imaging devices can detect small temperature differences. Cracks have different thermal resistances and thermal conductivities, and the resulting patterns can be more clearly distinguished through boundary highlighting preprocessing of infrared images. In [6], infrared light with a wavelength of 1,140 nm was projected from distances of 0.5 m and 1.3 m in order to detect cracks ranging in width from 0.2 mm to 1.5 mm, thereby improving performance.

The remainder of this paper is organized as follows. The next section analyses the causes of cracks and compares previous crack detection methods related to this study, highlighting their limitations. Section III presents a lightweight crack learning model that uses morphology-based preprocessing for infrared images. Section IV provides a detailed evaluation of the model's performance. Finally, Section V contains summaries and details of further studies.

II. SYSTEM MODEL

A. Types and Causes of Cracks

Concrete is a mixture of different materials and the chemical bonding between them to form a structure, so cracking is inevitable. Of course, not all cracks are problematic, but those that are larger than a certain size can cause problems that negatively affect not only the building itself, but also its users. The causes of cracking can be categorized into material, environmental, structural, and design factors as shown in Table 1.

To address these issues, many construction companies have dedicated quality control departments to identify the causes of cracking, and various types of research are being conducted. However, most of these efforts are limited to managing cracks after they occur or analyzing crack development in limited environments, which is not enough to minimize them in advance [7].

B. Non-destructive Test and Infrared Thermography

Non-destructive testing (NDT) is a type of testing that does not alter the original form or function of a material and is generally used to determine the nature, condition, and internal structure of a product without disassembling or destroying it. Among the various NDT techniques, infrared thermography is a NDT method that uses an infrared camera to diagnose abnormalities in the object, and uses the temperature signal of the object to detect defects that cause non-uniform temperature distribution in the object [1,8].

Table 1. Type of crack causes [7]

Classification	Contents
Material factors	- Cement: False setting, Heat of hydration - Aggregate: Low quality or Reactivity aggregate, Flour - Concrete: Chloride, Shrinkage, Bleeding
Environmental factors	- Temperature, Humidity: Freezing and thawing - Chemistry: Carbonation, Corrosion
Structural factors	- Load: Dead load, Live load, Wind load - Member size and kind
Construction factors	- Concrete: Mixing time, Poor compaction - Reinforced: Coating thickness, Reinforcement bad - Form: Swelling, Early removal

Thermal imaging cameras typically apply vibration or ultrasound to an object to capture a camera image synchronized to its frequency and are used primarily for non-destructive crack detection, where identification is more difficult than with conventional cameras.

C. Interpretive Image Processing Models

Image processing technology refers to the entire process of processing and analyzing images acquired from equipment, and consists of functions such as image input and output, preprocessing for digitization, segmentation, and defect detection. In order to effectively detect cracks, morphology techniques that detect cracks through morphological operations have been studied, and a method for detecting cracks by applying RGB channel values to fuzzy techniques to consider the saliency of cracks has been studied. [5,9,10]

In the traditional rule-based method, the user models a filter that removes noise from the image to effectively detect cracks. However, the disadvantage of this rule-based method is that new filter modeling is required depending on the focal length of the image, the effects of the shooting environment, the shooting quality, the resolution, and so on. In addition, the cameras developed to compensate for this are capable of capturing high resolution images, but are very expensive and therefore not economically viable [5].

To improve the overall performance of these interpretative crack detection methods, it is very important to know the statistical characteristics of the target image in advance in order to estimate the relevant parameters appropriately, which usually requires a rather complex implementation to identify additional features of the cracks and incorporate them into the design of the algorithm. These analytical approaches have the disadvantage of being inelastic compared to deep learning-based methods, which are less robust and can continuously improve their accuracy with additional data, especially when the statistical characteristics of the target images vary due to seasonal and climatic characteristics [11].

D. AI-based Image Processing

To compensate for the limitations of traditional techniques, the need for image analysis techniques using machine learning [12,13] and deep learning [3,14] has become more prominent in recent years, and related research has been actively conducted. In particular, there have been recent studies on how to inspect and analyze the appearance of large-scale aging infrastructure using drones equipped with video equipment using deep learning-based image processing techniques [5,15].

However, cracks that are difficult to find in existing images can be detected by using infrared images. In this study, we developed a deep learning method and image processing method that can effectively and quickly detect cracks and analyze the characteristics of cracks in concrete structures using infrared images, and verified the performance of the developed method for practical application.

E. Deep Learning-based Image Processing

For crack detection, it is necessary to segment the crack using image processing techniques. From an information perspective, the segmentation process converts low-level information (the original image) to higher-level information (the segmented image). It also removes unwanted information from the image, such as noise. Representative segmentation techniques include convolution-based segmentation and autoencoder-based segmentation [3,11].

III. METHODS AND RESULTS

A. Dataset

The dataset used in this experiment is a crack dataset consisting of 3,462 infrared camera images taken in 2024 and published on a computer vision dataset platform called Roboflow [16] aimed at object detection. It is a dataset of RGB images annotated in COCO format and resized to 640×640. It is also preprocessed with 50% horizontal flip and 50% vertical flip, and in this experiment we applied a combination of canny edge detection [17] and morphology [18] to verify the effectiveness of additional preprocessing.

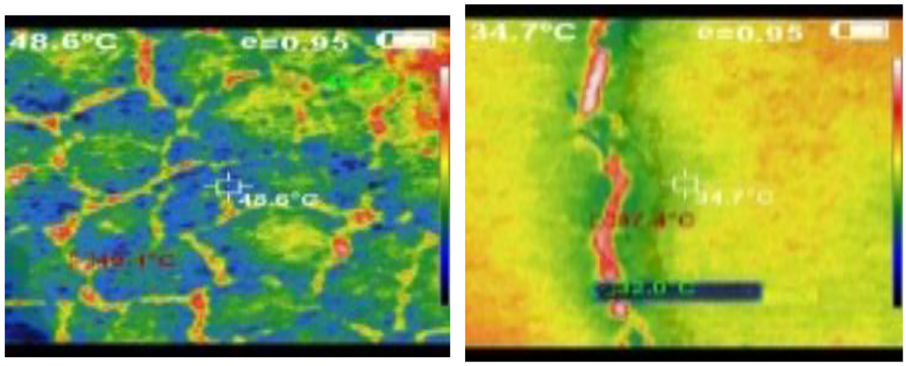
It is hard to find big data that has been collected separately using thermal infrared cameras to detect cracks under actual wallpaper. However, as Literature [7] shows, using infrared cameras is effective. It may be possible to obtain some crack data under the paper by measuring the old walls of our classrooms directly, if necessary.

Out of 3,462 images, 3,022 images were used as the training dataset for training, 293 images were used as the validation dataset for hyperparameter tuning, and 147 images were used as the test dataset for final performance evaluation of the model.

The dataset is classified into two types of cracks: alligator crack and long transverse crack. In the case of alligator crack, cracks are distributed in a wide area, such as the back shell of an alligator. Long transverse crack is a transverse crack and is distributed in a relatively narrower area than alligator crack, and the number of cracks in an image is smaller and the thickness of the crack is thicker.

In addition, each class is further divided into high, medium, and low according to the distinct degree, size, and number of cracks. Fig. 2 is an example image of the dataset.

For this experiment, we categorized the datasets according to the type, distinctiveness, size and number of



(a) (b)

Fig. 2. Example: alligator crack (a), long transverse crack (b).

cracks. The datasets were categorized for the following reasons. Firstly, real-world structures contain many different types of crack with various characteristics, each requiring different post-processing methods and levels of urgency for action. Therefore, more granular crack detection is meaningful, rather than simply identifying the presence or absence of cracks.

Secondly, in this paper, we considered the use of morphology and Canny-based preprocessing techniques alongside infrared detection techniques rather than simply detecting cracks. In particular, we refined the dataset further to study whether preprocessing techniques can have a significant effect when the boundaries of cracks are unclear, as with cracks labelled as "low" in the dataset.

Finally, we aimed to evaluate the reliability of light-weight models such as YOLOv5-nano under different crack conditions, and to study how well these models perform when utilizing preprocessing techniques compared to heavier models, particularly when detecting very small micro-cracks rather than clear cracks. This required a division in the dataset.

B. Deep Learning Models Used

1) YOLOv10 Model

YOLO [14] is a well-known deep learning model, and v10 [19] is the latest version of the YOLO object detection model, which is built on top of the Ultralytics Python package and performs well in a variety of computer vision tasks. YOLOv10 uses the Swin-Transformer as its backbone architecture, with strengths in feature representation, context capture, and accuracy, and uses EfficientNetV2 to balance efficiency and accuracy.

To improve the feature extraction ability, YOLO pre-trained the model with a large unlabeled dataset using a self-supervised learning method, and used gradient accumulation and mixed precision learning methods to allow training with larger batch sizes without memory constraints. In addition, the use of focal loss, which can focus more on difficult tasks and compensate for class imbalance, is one of the reasons why it was chosen as a suitable model for our experiment.

Non-maximum suppression (NMS) is an important post-processing technique in object detection that removes multiple redundant prediction boxes, leaving only the most reliable predictions. However, over-reliance on NMS in object recognition models negatively affects the model's ability to perform optimally, and YOLOv10 is designed to utilize consistent double mapping to enable training that does not rely on NMS, which has the advantage of reducing the model's inference speed. Fig. 3 shows the model structure of YOLOv10, and Fig. 4 shows performance metrics for several models on standard benchmarks such as COCO [19].

The YOLOv10 model receives images from the backbone and passes them through the path aggregation network (PAN) to fuse feature maps of different resolutions.

Additionally, the dual label assignment skill improves learning efficiency by applying a one-to-many method to improve recall and a one-to-one method to improve precision. The consistent matching metrics on the right use a new indicator combining confidence, prior and IoU.

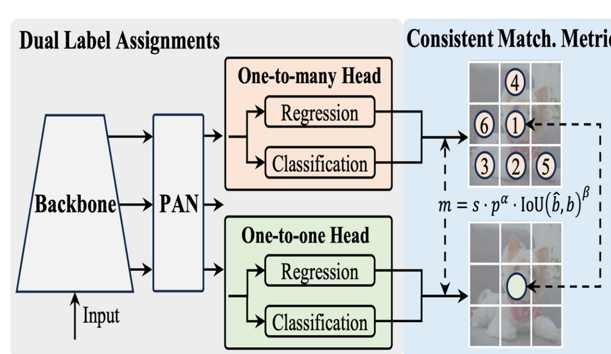


Fig. 3. YOLOv10 model structure diagram.

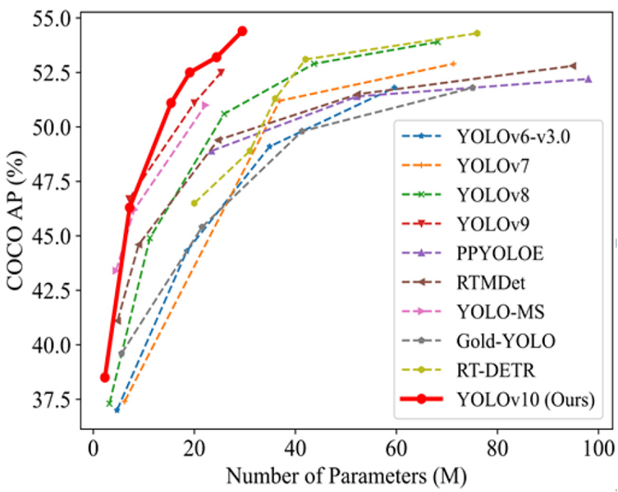


Fig. 4. Comparing the performance of different models.

The YOLOv10 model uses the same structure for learning and inference and is more efficient than existing YOLO models.

This graph illustrates the correlation between COCO AP (accuracy) and the number of parameters (model size) across different object detection models. The x-axis shows the number of parameters (in millions) and the y-axis shows the average precision (AP, in percentage) on the COCO benchmark.

YOLOv10 (red solid line) achieves significantly higher accuracy with fewer parameters than all the other models. It performs especially well in the lightweight model range (0–20M), achieving a balance of compactness and high accuracy. While models such as YOLOv8 and YOLOv9 perform well, they tend to be larger and heavier.

YOLOv10 demonstrates linear performance improvement as the model size increases. This indicates excellent scalability and efficiency in various deployment settings.

2) DETR Model

Detection with Transformer (DETR) [20] is a model developed by Facebook, which is designed by combining transformers, which are mainly used in natural language processing, with convolutional neural network (CNN) backbone and feed forward network (FFN), and has the structure as shown in Fig. 5. This is the first model to apply transformers to object detection, and it has the advantage of overcoming the limitation that CNN models extract features using only local information. As of August 2024, the DETR-based transformer model shows very high performance, achieving SOTA on datasets such as COCO test-dev, COCO minival, and COCO 2017 val. However, it is slow to train and requires a lot of data and computational resources.

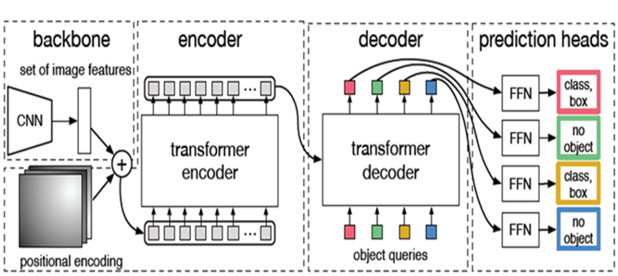


Fig. 5. DETR structure diagram.

This diagram illustrates the overall architecture of DETR. The input image is processed by the CNN backbone to extract feature maps, to which positional encoding is then added to retain spatial information. These encoded features are then passed to the Transformer encoder, which learns global relationships. A set of object queries is subsequently fed into the Transformer decoder. The decoder then combines the queries with the encoder outputs to infer the object instances. Each decoder output is then passed to a prediction head (FFN), which predicts either a class and a bounding box or indicates no object.

3) YOLOv5-nano Model

One of the goals of this experiment is to detect cracks using infrared imagery and, in particular, to detect cracks in real-time using low-end portable devices that do not have sufficient computational resources in the field. Furthermore, we believe that a lightweight model is needed to run an on-device AI model that processes data and detects cracks in real-time without relying on cloud-based servers. The aforementioned DETR has about 41 million parameters and YOLOv10 has 3 million parameters, while YOLOv5-nano [21] has about 1.7 million parameters, making it a relatively lightweight model.

C. Preprocessing Algorithms

1) Canny Edge Detection Algorithm

Canny edge detection is a technique used in image processing before the advancement of deep learning network technology. It is an algorithm that extracts edges by denoising using a Gaussian filter and exploring regions with high gradient values (Fig. 6). It helps identify the boundaries of objects in an image, can detect a wide range of edges with high accuracy, and provides clean and thin edges [17].

The Canny algorithm comprises several key stages—noise reduction, gradient calculation, NMS, and edge tracking by hysteresis. Firstly, a Gaussian filter is applied to the image to remove unwanted noise and smooth it out. Next, the image's intensity gradients are calculated using techniques such as the Sobel operator, which highlights regions with strong spatial derivatives. Next, NMS is performed to refine the edges and retain only the local maxima of the gradient magnitude, ensuring the resulting edges are sharp and precise. Finally, hysteresis thresholding connects weak edges to strong ones based on dual-threshold

(a) (b)

Fig. 6. Example image of Canny edge detection: (a) original and (b) Canny edge detection applied.

values, enhancing edge continuity while reducing false detections.

In this study, we conducted experiments to tune the parameters of the Canny edge detection algorithm and determine the most effective settings for infrared crack images. Based on preliminary tests using various threshold values, we set the lower threshold to 50 and the upper threshold to 80. Additionally, we blended the final edge-detected result with the original infrared image using an alpha value of 0.6. This alpha blending preserved the thermal texture while enhancing crack contours.

These parameter values were selected empirically: a lower threshold of 50 effectively suppressed minor noise, and an upper threshold of 80 allowed the algorithm to respond to moderately strong edges that corresponded well to actual crack boundaries in our dataset. The alpha value of 0.6 achieved a good balance between the visual appearance and structural integrity of the original image and the edge-enhanced overlay, facilitating the identification of crack regions by the deep learning model.

The dataset of this experiment is a set of photo taken by an infrared camera, which is not a high-end camera and contains a lot of noise. In the case of cracks, which are objects to be detected, capturing the boundaries of cracks has a greater impact on determining whether it is an object than other factors such as texture and color. Therefore, it is expected that the performance of the model can be improved if preprocessing is done to highlight the boundaries using canny edge detection.

Infrared images, in particular, often lack strong brightness or color contrast in cracks, making the clear boundary definition provided by edge detection algorithms like Canny crucial. By emphasizing the structural transitions in temperature distribution, Canny edge detection helps distinguish subtle crack patterns that might otherwise be overlooked. Furthermore, applying Canny edge detection to an image prior to feeding it into a deep learning model can enhance the model's ability to learn meaningful representations from limited training data.

2) Morphology Algorithm

Morphology is a technique for simplifying the results of image segmentation to make it easier to find objects of interest. Morphology includes dilation operations, which fill holes in an object and connect two nearby objects, and erosion operations, which erode the boundaries of an object and remove small bumps to reduce the size of the object [18].

The opening operation, which performs an erosion operation on an object and then applies a dilation operation, can be used to remove pixels that are small and contained within the object, and the closing operation, which performs a dilation operation and then applies another erosion operation, can be used to fill small holes or gaps within the object. This can provide a basic denoising effect and emphasize the boundaries of the object (Fig. 7). In particular, the cracks in our experimental dataset are often intersected by different cracks, and the morphology technique helps to separate each intersected crack into different objects for detection.

In our experiment, we used morphology as a preprocessing step to improve the shape and continuity of crack features in thermal infrared images. Due to the characteristics of cracks, such as their thin, elongated shape and frequent discontinuities, we empirically selected a 3×3 kernel size with an elliptical structuring element. This balances the need to connect fragmented crack segments while minimizing the risk of merging unrelated nearby structures. We also applied the closing operation, followed by a single iteration of opening, to fill small gaps and remove isolated noise pixels. These hyperparameters were specifically chosen to address the noisy nature of low-resolution infrared images, in which cracks may be faint or partially obscured. By emphasizing linear continuity and suppressing minor background interference, the morphological preprocessing helped generate clearer input images for subsequent deep learning models, particularly lightweight architectures such as YOLOv5-nano, which benefit from enhanced boundary cues.

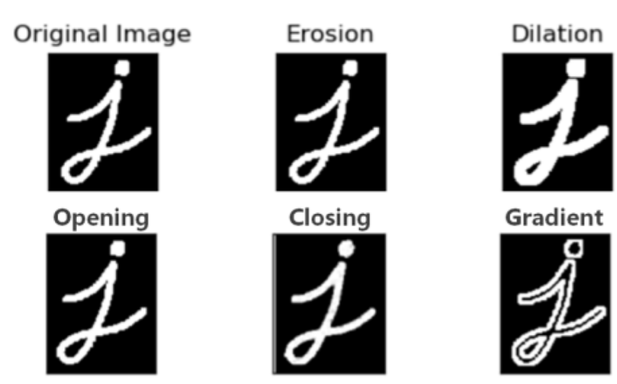
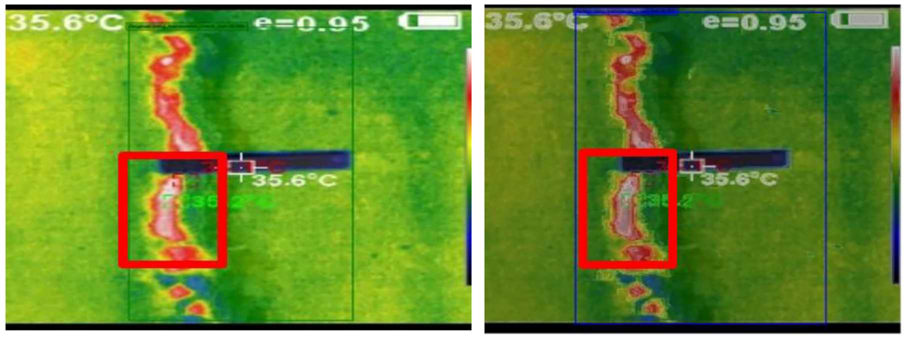


Fig. 7. Example image of morphology operation.



(a) (b)

Fig. 8. (a) Original (a) and Canny edge detection applied (b).

IV. PERFORMANCE EVALUATION

A. Image Preprocessing Results

In this experiment, we tried three preprocessing methods: Canny edge detection, morphology, and mixed use to test whether image preprocessing can improve performance. All the preprocessors were superimposed on the original image after preprocessing to emphasize the features that we want to emphasize by preprocessing, while preserving the texture, color information, etc., of the original image as much as possible (Figs. 8 and 9).

As can be seen in Fig. 8, in which the Canny edge detection algorithm has been applied to an example image, the boundaries of the cracks are clearly highlighted compared to the original image, and the fine patterns are more pronounced.

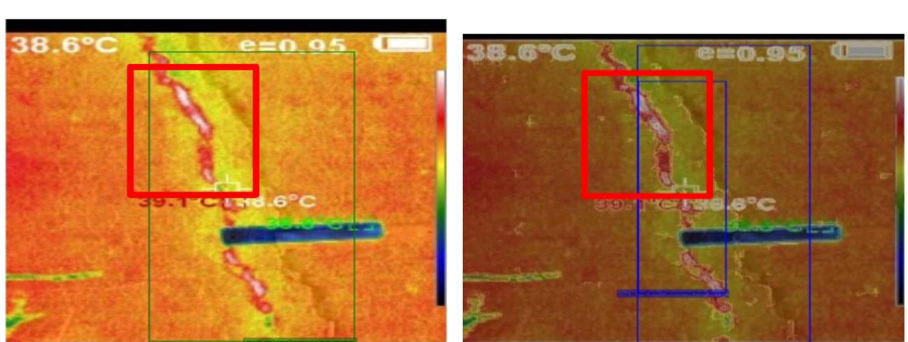
Due to the nature of infrared images, the boundaries are extracted based on differences in heat distribution. This means that fine crack patterns that were previously blurred can now be seen more clearly. This effect is more pronounced in lower-resolution infrared images, and there is a possibility of some false positives due to the boundaries being overemphasized.

Fig. 9 shows an example of an image that has been processed using morphology and the Canny edge detector, and then stitched back together with the original image.

The morphology operation emphasizes the continuity of the cracks in the original image. The closing operation naturally connects fine cracks that appeared disconnected in the original image, while the opening operation effectively removes background noise. This approach is effective in the crack detection stage, but is particularly useful when classifying cracks by refining the labels after detection. Notably, this approach maintains a level of computational efficiency that enables real-time processing on low-performance embedded equipment.

B. DETR Experimental Results

For DETR, we used the pretrained ResNet-50 backbone. The model was trained for 60 epochs with a batch size of 4, using AdamW (lr = 1e-4, weight decay = 1e-4). After training the model, the learning progress graphs are shown in Figs. 10 and 11. The training loss was 0.09 and the validation loss was 0.47, and compared to the CNN-based model without transformers, more training was required to achieve training stabilization. The training time



(a) (b)

Fig. 9. Morphology applied image (a) and blending applied image (b).

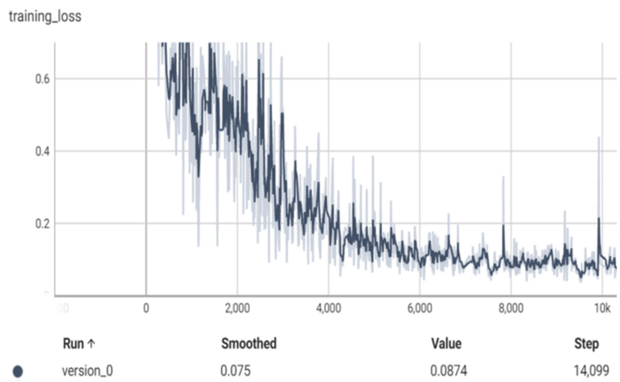


Fig. 10. Training loss of DETR learning progress graphs.

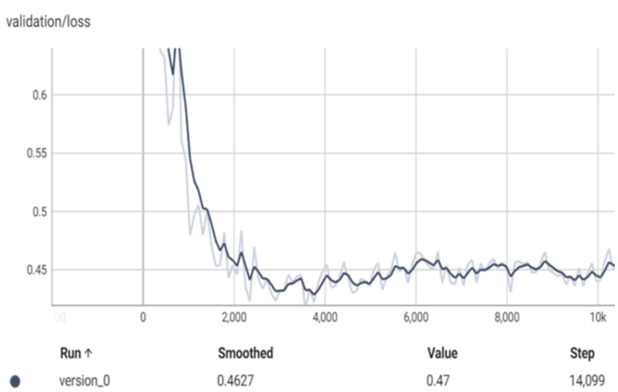


Fig. 11. Validation loss of DETR learning progress graphs.

required to achieve similar performance generally varied with hyperparameters such as batch size, epoch, and optimizer, but was four times as long as in YOLOv10.

The DETR model has approximately 41.5 million parameters and is expected to require a high-end multi-core CPU and at least 16 GB of RAM. This is only possible with fixed infrared imagers that have sufficient computing resources, so a lighter model is appropriate.

Table 2 shows the results of the DETR model. It achieves relatively high recall but low precision. However, false positives can occur in complex infrared patterns. Additionally, the relatively low mAP50-95 value indicates

Table 2. Test results of DETR model

Metric	Value	Description
mAP50	0.9664	Average precision across IoU 0.50
mAP50-95	0.7821	AP at IoU threshold 0.50–0.95
Precision	0.9402	Ratio of true positives to all positives
Recall	0.9834	Ratio of true positives to all actual positives

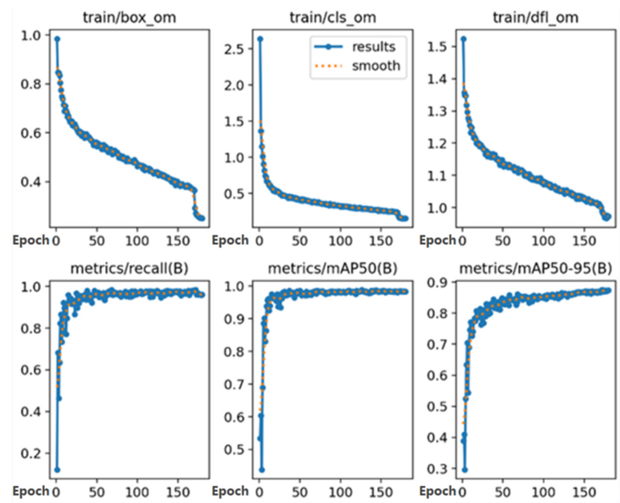


Fig. 12. Learning progress graphs of YOLOv10.

poor generalization performance when it comes to cracks of different sizes and angles.

C. YOLOv10 Experimental Results

YOLOv10 was trained with pretrained weights (yolov10n.pt) for 180 epochs with a batch size of 32, optimized by AdamW (lr ≈ 8.3e-4). After training the model, the learning progress graphs of YOLOv10 are shown in Fig. 12. The experimental results showed excellent performance with precision 0.981, mAP50 0.985 and processing time 5.8 ms, but it has about 2.7 million parameters and is lighter than DETR with 41 million parameters, but it has 1.5 times more parameters than YOLOv5-nano with about 1.7 million parameters. This is probably feasible in real-time with high-end infrared devices. For low-end portable devices, a much lighter model is appropriate.

D. YOLOv5-nano Experimental Results

YOLOv5-nano has the smallest number of parameters among the models in this experiment with 1.7 million parameters, and the processing speed is 8.0 ms, which is 3.2 ms slower than YOLOv10, but is considered fast enough for real-time processing. Precision was 0.956 and mAP50 was 0.970, which is a difference of about 1% compared to YOLOv10. In addition, preprocessing techniques were not effective in DETR or YOLOv10 with many parameters because the model already had enough performance, but YOLOv5-nano could achieve a performance improvement of 1%–3%.

YOLOv5-nano was trained from scratch for 150 epochs with a batch size of 16, using SGD (lr = 0.01, momentum = 0.937, weight decay = 0.0005). Table 3 shows the experimental results of preprocessing the YOLOv5-nano

Table 3. YOLOv5-nano results with none, Canny, morphology, and Canny & morphology

	None	Canny	Morphology	Canny & morphology
Train box loss	0.0136	**0.0134**	0.0135	0.0137
Train obj loss	0.0104	0.0105	**0.0103**	0.0106
Train ds loss	0.0035	0.0034	0.0034	**0.0033**
Precision	0.9556	0.9646	**0.9736**	0.962
Recall	0.9748	0.9748	**0.977**	0.9695
mAP50	0.9702	**0.9794**	0.9779	0.9733
mAP50-95	0.7664	0.786	**0.7861**	0.7716
Val box loss	0.0043	0.0043	**0.0042**	0.0044
Val obj loss	**0.0027**	0.0029	0.0028	0.0029
Val ds loss	0.0006	**0.0004**	**0.0004**	0.0005

The bold font indicates the best performance in each test.

model with none, canny, morphology, and canny & morphology. The effect of preprocessing on train loss and validation loss was very small, but precision, recall, mAP50, and mAP50-95, which are direct indicators of how accurately the crack was detected, showed a positive effect of preprocessing. Morphology had the best effect, with a performance improvement of 1.9%–2.6% compared to no preprocessing. Preprocessing did not produce dramatic gains because performance was already high (95%–97%) without preprocessing, but it is possible that preprocessing could be more effective on more difficult tasks.

Table 4 shows a comparison between YOLOv10 and YOLOv5-nano trained with morphology preprocessing. YOLOv5-nano uses 929,473 fewer parameters than YOLOv10, which is 65.53% of the weight of YOLOv10, making it a lighter model. As a result, the inference time is also 2 ms less than YOLOv10 at 5.3 ms, which is a big advantage in the real world where speed and efficiency are critical. YOLOv5-nano has 137% faster processing speed with fewer parameters than YOLOv10, but accuracy is within 1%. In particular, for crack detection, the task is more critical than misclassifying something that is not a crack as a crack, so recall is more important than precision. YOLOv5-nano is 1% lower in precision than YOLOv10, but 1% higher in recall, so it makes sense to use YOLOv5-nano for this experiment.

In the context of the present study, the DETR and YOLOv10 models are characterized by a large number of parameters and intricate architectures. These features enable them to autonomously acquire significant visual attributes, such as boundaries, textures and patterns, from imagery. Specifically, the DETR model employs its transformer-based structure to learn global information, whereas the YOLOv10 model combines Swin-Transformer and EfficientNet to facilitate powerful feature extraction. Consequently, these models can learn the crack contour and structural shape adequately without the need for preprocessing. Therefore, the preprocessing methods employed in this experiment—Canny and morphology, which focus on boundary extraction—may not have a significant impact.

The infrared image dataset used in our experiments is characterized by low resolution, noise, and frequently unclear cracks. Using preprocessing techniques to enhance the clarity of the initial visual information could influence how the models learn when operating on these images. However, DETR and YOLOv10 are already highly expressive models, so any additional information from preprocessing is likely to be redundant or ignored.

Table 4. Comparison of YOLOv10 and YOLOv5-nano trained with morphology preprocessing

	YOLOv10	YOLOv5-nano + Morphology	YOLOv5-nano/v10
Parameter	2,696,756	**1,767,283**	65.53%(+)
Inference (ms)	5.3	**3.3**	160.60%(+)
Precision	**0.981**	0.974	99.29%(-)
Recall	0.967	**0.977**	101.03%(+)
mAP50	**0.985**	0.978	99.29%(-)

The bold font indicates the best performance in each test.

Conversely, the comparatively simple and lightweight architecture of the YOLOv5-nano model limits its ability to capture intricate image details during training. In such cases, applying preprocessing techniques to accentuate crack boundaries or shapes can provide the model with more direct information for learning purposes, thereby improving precision and recall. In summary, the impact of preprocessing diminishes in more complex models, while its effect is amplified in less complex ones.

V. CONCLUSION

In this study, various image preprocessing methods and a YOLO-based lightweight deep learning model were experimentally analyzed to improve the performance of crack detection using infrared thermal imaging. By applying Canny edge detection, morphology, and mixed methods, we found that the performance is most dominant when using morphology over the original image, which is believed to be the result of effectively highlighting the boundaries and shape of the crack.

In particular, the preprocessing method showed unsatisfactory performance for complex and heavy models such as DETR and YOLOv10, but was able to provide a 1%–3% performance improvement for light models. In addition, a comparison between YOLOv10 and YOLOv5-nano showed that the YOLOv5-nano model used with the preprocessing method had a faster processing speed and a similar level of accuracy with a smaller number of parameters (1.7 million). Therefore, this study demonstrates that lightweight deep learning models for infrared thermal crack detection can perform similarly to heavier models when appropriate image preprocessing techniques are applied, which is expected to be particularly beneficial in the field where low-end detection devices are often used.

Nevertheless, the present study is based on a single dataset, which has limitations in terms of both data diversity and generalizability. Furthermore, the dataset's constraints prevented testing in environments where infrared could be most effectively utilized for crack detection. These include instances where cracks are not directly visible from the exterior, for example when they are concealed within wallpaper or situated within the structure itself.

Future research will expand the dataset to include different types of structures, materials, and lighting conditions. NDT technologies, such as ultrasonic waves and X-rays, effectively detect cracks under wallpaper or paint. These methods can be used to verify and compare the predicted results of the proposed model with those of real structures externally. Using multiple sensor data to predict crack depth can improve the model's robustness. Additionally, research should be conducted on implementing and optimizing a real-time, on-device detection system based on a lightweight model. These efforts will increase the model's practical applicability and ensure its reliability in construction and safety.

CONFLICT OF INTEREST

The authors have declared that no competing interests exist.

REFERENCES

1. J. Y. Jung, H. J. Yoon, H. W. Cho, and H. S. Yang, "a study on temperature characteristics of various depth using infrared thermography," *Journal of the Korea Academia-Industrial cooperation Society*, vol. 19, no. 3 pp. 83-89, 2018. https://doi.org/10.5762/KAIS.2018.19.3.83
2. S. H. Kim, K. Kim, and N. Kim, "A applicability study on the asphalt concrete pavement condition index in narrow regional roads using road crack," *Journal of the Korean Society of Civil Engineers*, vol. 38, no. 3, pp. 467-475, 2018. https://doi.org/10.12652/Ksce.2018.38.3.0467
3. S. D. Nguyen, T. S. Tran, V. P. Tran, H. J. Lee, M. J. Piran, and V. P. Le, "Deep learning-based crack detection: a survey," *International Journal of Pavement Research and Technology*, vol. 16, no. 4, pp. 943-967, 2023. https://doi.org/10.1007/s42947-022-00172-z
4. N. I. M. Yusof, A. Sophian, H. F. M. Zaki, A. A. Bawono, A. H. Embong, and A. Ashraf, "Assessing the performance of YOLOv5, YOLOv6, and YOLOv7 in road defect detection and classification: a comparative study," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 1, pp. 350-360, 2024. https://doi.org/10.11591/eei.v13i1.6317
5. A. R. Kim, D. Kim, Y. S. Byun, and S. W. Lee, "Crack detection of concrete structure using deep learning and image processing method in geotechnical engineering," *Journal of the Korean Geotechnical Society*, vol. 34, no. 12, pp. 145-154, 2018. https://doi.org/10.7843/kgs.2018.34.12.145
6. A. Karpf, M. Selig, A. Alchaar, and M. Iskander, "Detection of cracks in concrete using near-IR fluorescence imaging," *Scientific Reports*, vol. 13, no. 1, article no. 18880, 2023. https://doi.org/10.1038/s41598-023-45917-3
7. M. J. Kim, "A study of crack shape analysis based on properties of member in apartment exterior wall," *Korean Journal of Construction Engineering and Management*, vol. 16, no. 3, pp. 143-151, 2015. http://doi.org/10.6106/KJCEM.2015.16.3.143
8. S. Bagavathiappan, B. B. Lahiri, T. Saravanan, J. Philip, and T. Jayakumar, "Infrared thermography for condition monitoring: a review," *Infrared Physics & Technology*, vol. 60, pp. 35-55, 2013. https://doi.org/10.1016/j.infrared.2013.03.006
9. Y. Kim, "Development of crack recognition system for concrete structure using image processing method," *Journal of the Korea Institute of Information Technology*, vol. 14, no. 10, pp. 163-168, 2016. http://doi.org/10.14801/jkiit.2016.14.10.163
10. H. S. Park, "Performance analysis of the tunnel inspection system using high speed camera," *Journal of the Korea Institute of Information Technology*, vol. 11, no. 4, pp. 1-6, 2013. https://www.kci.go.kr/kciportal/ci/sereArticleSearch/

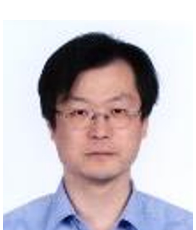
ciSereArtiView.kci?sereArticleSearchBean.artiId=ART001761707
11. J. Ha, K. Park, and M. Kim, "A development of road crack detection system using deep learning-based segmentation and object detection," *Journal of Society for e-Business Studies*, vol. 26, no. 1, pp. 93-106, 2021. https://doi.org/10.7838/jsebs.2021.26.1.093
12. K. Myagila and H. Kilavo, "A comparative study on performance of SVM and CNN in Tanzania sign language translation using image recognition," *Applied Artificial Intelligence*, vol. 36, no. 1, article no. 2005297, 2022. https://doi.org/10.1080/08839514.2021.2005297
13. S. Byun, "Composite context-based offloading scheme for IoT fault diagnosis," *Journal of Next-generation Convergence Technology Association*, vol. 7, no. 5, pp. 762-773, 2023. https://doi.org/10.33097/JNCTA.2023.07.05.762
14. M. Hussain, "YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection," *Machines*, vol. 11, no. 7, article no. 677, 2023. https://doi.org/10.3390/machines11070677
15. S. Cho, B. Kim, and Y. I. Lee, "Image-based concrete crack and spalling detection using deep learning," *KSCE Magazine*, vol. 66, no. 8, pp. 92-97, 2018. https://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE07504737
16. Roboflow, "infrared model: computer vision model", 2024 [Online]. Available: https://universe.roboflow.com/object-detection-lmq58/infrared-model.
17. J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, 1986. https://doi.org/10.1109/TPAMI.1986.4767851
18. P. Soille, *Morphological Image Analysis: Principles and Applications*. Heidelberg, Germany: Springer, 1999. https://doi.org/10.1007/978-3-662-03939-7
19. A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, and J. Han, "YOLOv10: real-time end-to-end object detection," 2024 [Online]. Available: https://arxiv.org/abs/2405.14458.
20. N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," 2020 [Online]. Available: https://arxiv.org/abs/2005.12872.
21. J. Wang, Y. Chen, Z. Dong, and M. Gao, "Improved YOLOv5 network for real-time multi-scale traffic sign detection," *Neural Computing and Applications*, vol. 35, no. 10, pp. 7853-7865, 2023. https://doi.org/10.1007/s00521-022-08077-5

Joonho Byun https://orcid.org/0009-0001-7874-7281

Joonho Byun receivSed his B.S. degree in Electronic Engineering from Kookmin University, Seoul, Korea, in 2022. From March 2023 to January 2024, he was an AI researcher with the Industrial AIoT research center at Anyang University, Anyang, Korea. He is in an M.S. at the Graduate School of Artificial Intelligence at Sogang University, Seoul, Korea. He is a recipient of best paper award in 2023. His main research interests include computer vision, IoT, machine learning, deep learning, and enhanced vision correction.

Jeong-Joon Kim https://orcid.org/0000-0002-0125-1907

Jeong-Joon Kim received his B.S. and M.S. degrees in Computer Science at Konkuk University in 2003 and 2005, respectively. In 2010, he received his Ph.D. in at Konkuk University. He is currently a professor in the Department of Software, Anyang University, South Korea. His research interests include database systems, bigdata, artificial intelligence (AI), semantic web, geographic information systems (GIS) and ubiquitous sensor network (USN), etc.



Siwoo Byun https://orcid.org/0000-0002-4887-6875

Siwoo Byun received his M.S. and Ph.D. degrees in Computer Science from KAIST, South Korea, in 1991, and 1999, respectively. Since March 2000, he has been a professor in the Department of Software, Anyang University, South Korea. He Holds a Ph.D. degree with a specialization in database transaction analysis. His research areas are IoT data processing, IoT machine learning, embedded system, big database, flash memory database, medical healthcare and sensor pattern recognition. He is a recipient of best research scholar award and best paper award. His new research interests include enhanced image/signal processing, deep learning, and road facility recognition, IoT storage system, and compound sensor offloading.