

Deep Learning Based Automated Crack Detection Using the SDNET2018 Dataset

Long Qiao

Construction Technology and Management, School of Built Environment
Ferris State University, Big Rapids, MI 49307, USA
Corresponding Author: Long Qiao

ABSTRACT

Reliable crack detection can improve the consistency and efficiency of bridge inspection. This study evaluates a ResNet18 binary classifier for distinguishing cracked and non-cracked concrete images in the SDNET2018 dataset, with particular attention to class imbalance and training duration. The original dataset contained approximately 8,484 cracked images and 47,608 non-cracked images. A balanced dataset was constructed by augmenting cracked samples, producing 34,396 images per class for training, 6,070 per class for validation, and approximately 7,142 per class for testing. On the original imbalanced test set, the model achieved 93.02% accuracy but only 59.54% recall for cracked images. With balanced data, cracked-class recall increased to 89.62% after 10 epochs, while overall accuracy was 92.75%. Extending training to 20 epochs increased overall accuracy to 93.21% and cracked-class F1-score to 92.95%, although cracked-class recall remained essentially unchanged at 89.55%. These results show that accuracy alone can conceal poor defect sensitivity and that class balancing substantially improves crack detection. The findings support the use of balanced training data for automated inspection research while indicating the need for validation on independent field imagery before operational deployment.

Keywords: bridge inspection; crack detection; class imbalance; convolutional neural network; ResNet18; SDNET2018.

Date of Submission: 13-09-2026

Date of acceptance: 24-09-2026

I. INTRODUCTION

Bridges are essential components of transportation networks, and their deterioration can result from material aging, environmental exposure, repeated loading, and construction or maintenance deficiencies. Visual inspection remains the primary means of identifying surface damage, but the process is labor-intensive, depends on inspector judgment, and may expose personnel to traffic and difficult access conditions. Computer vision and deep learning offer a way to screen large image collections consistently and to direct inspectors toward locations that require closer examination.

The performance of image-based crack detection depends on the quality and composition of the training data. SDNET2018 is a widely used concrete-crack dataset that includes bridge deck, wall, and pavement images, but its cracked and non-cracked classes are strongly imbalanced. A classifier trained without addressing this imbalance may achieve high overall accuracy by favoring the non-cracked class while missing a substantial proportion of cracks. This study therefore examines how class balancing affects a ResNet18 classifier and whether extending training from 10 to 20 epochs materially changes its performance. The analysis emphasizes cracked-class recall and F1-score because missed cracks are more consequential than the aggregate accuracy value alone suggests.

II. LITERATURE REVIEW

Deep learning has expanded the capabilities of image-based bridge inspection. Convolutional neural networks can learn surface features directly from images and have been applied to defect classification, localization, and segmentation. Their performance, however, remains sensitive to class distribution, image quality, surface texture, illumination, and the similarity between training data and field conditions.

Cardellicchio et al. (2022) evaluated deep-learning architectures for image-based detection and classification of structural defects in reinforced-concrete bridges. Their work illustrates the value of transfer learning for recognizing multiple defect categories and also identifies practical difficulties caused by limited data and visual variability. These considerations support the use of a residual network in the present study and motivate explicit attention to class balance.

Chun et al. (2022) extended image-based inspection beyond classification by combining a convolutional encoder with a recurrent decoder to generate descriptions of bridge damage. Such systems could

eventually supplement defect classifiers by communicating the type and apparent location of observed damage. Their usefulness depends on reliable visual recognition, which makes the quality and balance of the underlying training data important.

Recent research also addresses the quality of bridge-inspection data. Zhang et al. (2025) examined the use of machine learning to improve data quality in bridge inspection, reinforcing the broader point that model performance depends on the consistency and reliability of its inputs. For crack-detection systems, blur, shadows, surface stains, and texture variation can create patterns that resemble cracks or conceal narrow defects.

Swamy et al. (2024) investigated image enhancement and machine-learning methods for bridge inspection. Image enhancement may improve the visibility of surface features, but preprocessing does not eliminate the need for representative labeled data. Models intended for inspection must distinguish uncommon defects even when non-defect images dominate the available dataset.

The literature shows that deep networks can support bridge-defect recognition, but reported performance depends heavily on dataset design and evaluation procedures. Residual networks provide an established basis for image classification, while augmentation can increase variation in underrepresented classes. Metrics must also be selected carefully: a high overall accuracy may coexist with an unacceptable rate of missed defects.

Class imbalance is therefore a central concern in safety-related image classification. When non-cracked images dominate training and testing, overall accuracy gives disproportionate weight to the majority class. Per-class precision, recall, F1-score, and confusion matrices provide a more informative assessment. Crack recall is particularly important because it measures the proportion of cracked images that the model identifies.

This study contributes a controlled comparison of one ResNet18 architecture under an imbalanced condition and two balanced conditions. It quantifies the change in cracked-class performance after balancing and compares 10- and 20-epoch training runs on the balanced dataset. The objective is not to establish field readiness, but to determine how strongly dataset composition affects the reported performance of a compact convolutional model.

III. METHODOLOGY

SDNET2018 contains labeled 256×256 -pixel images of concrete bridge decks, walls, and pavements (Maguire et al., 2018). Each image is assigned to a cracked or non-cracked class. The original collection used in this study contained approximately 8,484 cracked images and 47,608 non-cracked images. To reduce class imbalance, cracked samples were augmented using rotations, horizontal and vertical flips, and random image transformations. The balanced dataset contained 34,396 images per class in the training split, 6,070 per class in the validation split, and approximately 7,142 per class in the test split. Figure 1 presents representative SDNET2018 images, and Figure 2 illustrates the augmentation process.

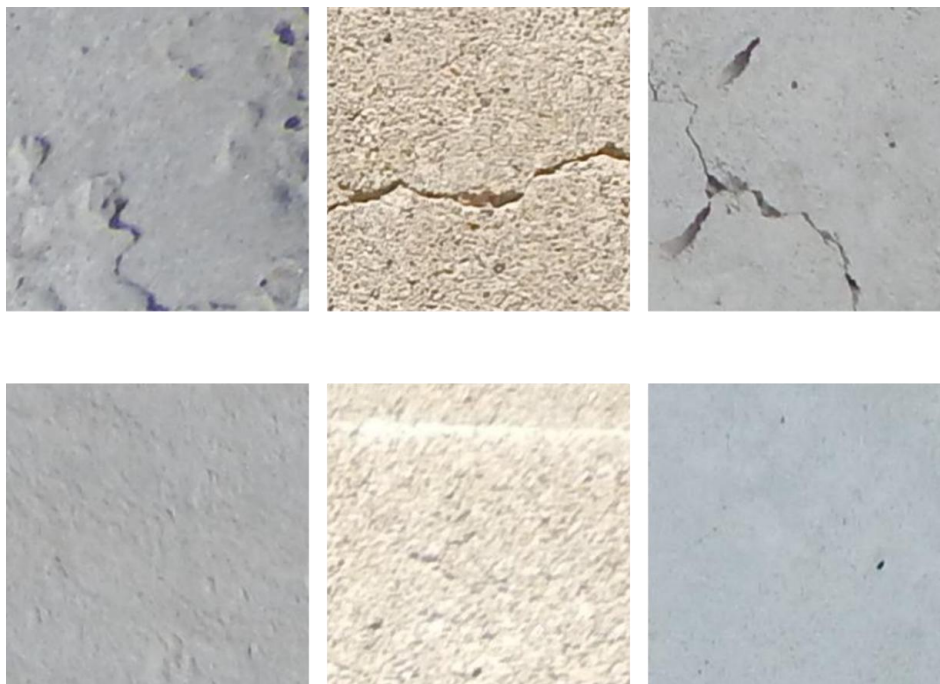


Figure 1: Sample images from the SDNET2018 dataset

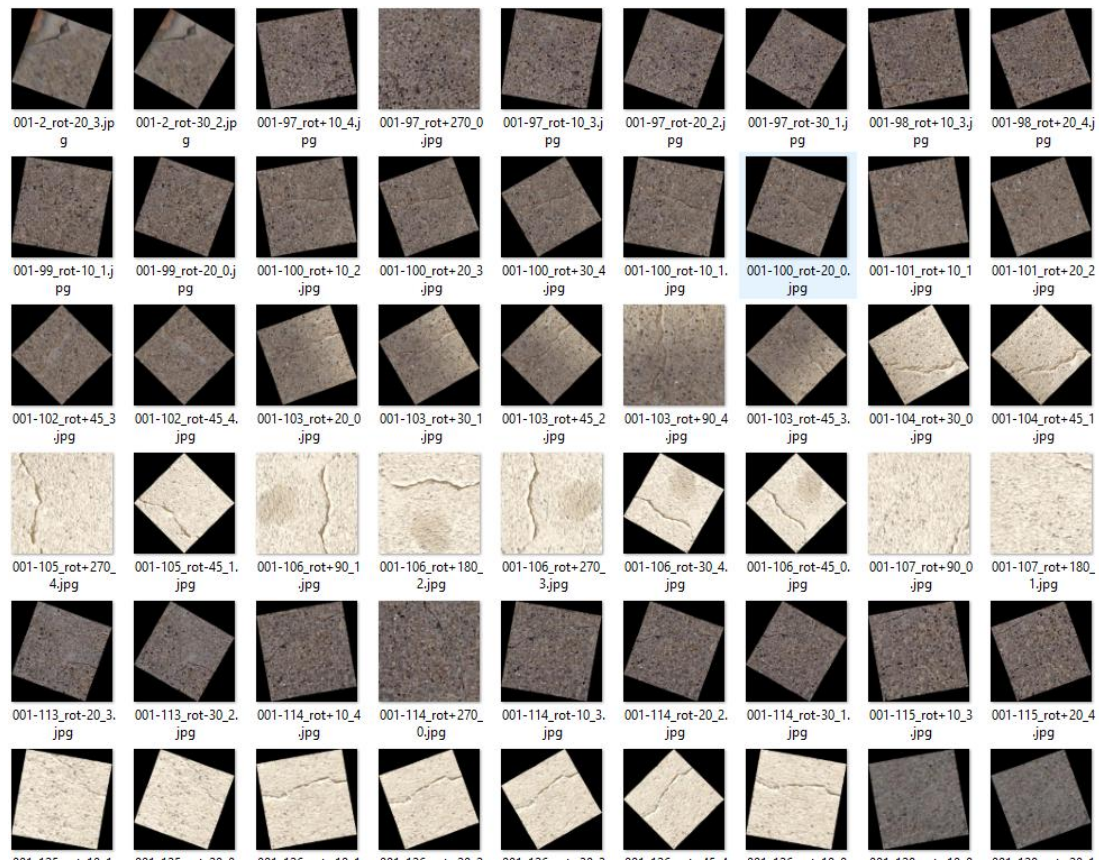


Figure 2: Examples of data augmentation

A ResNet18 convolutional neural network was used for binary classification. The final fully connected layer was modified to produce two outputs representing cracked and non-cracked surfaces. Training used a batch size of 32, the Adam optimizer, and cross-entropy loss. Three experimental conditions were evaluated: the original imbalanced dataset trained for 10 epochs, the balanced dataset trained for 10 epochs, and the balanced dataset trained for 20 epochs.

Performance was evaluated using accuracy, precision, recall, F1-score, and confusion matrices. Accuracy measures the proportion of all correct predictions, whereas class-specific recall measures the proportion of actual images in a class that were correctly identified. Because the practical concern is failure to identify damage, the interpretation gives particular weight to recall and F1-score for the cracked class.

IV. RESULTS AND DISCUSSION

The three experimental conditions produced similar overall accuracy values but substantially different cracked-class recall. This difference demonstrates why aggregate accuracy should be interpreted together with class-specific metrics when the class distribution is unequal.

4.1 Performance on the Imbalanced Dataset

The model trained on the original imbalanced dataset achieved 93.02% accuracy (Table 1). This value masked weak sensitivity to cracks: the model correctly classified 758 of 1,273 cracked images, corresponding to a cracked-class recall of 59.54%. It misclassified 515 cracked images as non-cracked, compared with only 72 non-cracked images classified as cracked. The resulting cracked-class F1-score was 72.09%. The confusion matrix therefore indicates a strong majority-class bias despite the apparently high accuracy.

Table 1: Performance after 10 epochs on the imbalanced dataset

Class	Precision	Recall	F1-Score	Support
NonCracked	0.9321	0.9899	0.9601	7141
Cracked	0.9133	0.5954	0.7209	1273

Overall accuracy 0.9302

Confusion Matrix

- Correctly classified non-cracked images 7,069
- Correctly classified cracked images 758
- Non-cracked images classified as cracked 72
- Cracked images classified as non-cracked 515

4.2 Performance after 10 Epochs on the Balanced Dataset

Balancing the dataset substantially improved crack sensitivity. After 10 epochs, the model achieved 92.75% accuracy and a cracked-class recall of 89.62% (Table 2). The cracked-class F1-score increased from 72.09% in the imbalanced condition to 92.51%, while non-cracked recall remained high at 95.87%. The model correctly classified 6,400 of 7,141 cracked images and 6,847 of 7,142 non-cracked images. Relative to the imbalanced condition, balancing increased cracked-class recall by 30.08 percentage points.

Table 2: Performance after 10 epochs on the balanced dataset

Class	Precision	Recall	F1-Score	Support
NonCracked	0.9023	0.9587	0.9297	7142
Cracked	0.9559	0.8962	0.9251	7141

Overall accuracy 0.9275

Confusion Matrix

- Correctly classified non-cracked images 6,847
- Correctly classified cracked images 6,400
- Non-cracked images classified as cracked 295
- Cracked images classified as non-cracked 741

4.3 Performance after 20 Epochs on the Balanced Dataset

Extending balanced training to 20 epochs increased overall accuracy from 92.75% to 93.21% and increased the cracked-class F1-score from 92.51% to 92.95% (Table 3). Non-cracked recall improved from 95.87% to 96.86%, and false-positive crack predictions decreased from 295 to 224. Cracked-class recall changed only from 89.62% to 89.55%, while cracked images classified as non-cracked increased slightly from 741 to 746. The longer run therefore improved overall accuracy primarily by reducing false-positive crack predictions; it did not reduce the number of missed cracked images.

Table 3: Performance after 20 epochs on the balanced dataset

Class	Precision	Recall	F1-Score	Support
NonCracked	0.9027	0.9686	0.9345	7142
Cracked	0.9662	0.8955	0.9295	7141

Overall accuracy 0.9321

Confusion Matrix

- Correctly classified non-cracked images 6,918
- Correctly classified cracked images 6,395
- Non-cracked images classified as cracked 224
- Cracked images classified as non-cracked 746

Key comparison findings

- Overall accuracy increased by 0.46 percentage points from 10 to 20 epochs.
- Non-cracked recall increased from 95.87% to 96.86%.
- Cracked recall remained nearly constant, changing from 89.62% to 89.55%.
- False-positive crack predictions decreased, while false-negative crack predictions remained similar.

4.4 Comparison of the Balanced Training Runs

Tables 4 and 5 compare the balanced 10- and 20-epoch runs. The additional training produced modest gains in accuracy and F1-score, but the two models had nearly identical cracked-class recall. For inspection screening, this distinction matters: the 20-epoch model generated fewer false alarms, yet it did not identify a larger proportion of cracked images. The results support dataset balancing as the principal improvement observed in this experiment, whereas the benefit of extending training was comparatively small.

Table 4: Performance metrics for the balanced 10- and 20-epoch runs

Metric	10 Epochs	20 Epochs	Change
Accuracy	0.9275	0.9321	↑ +0.0046
Precision (Cracked)	0.9559	0.9662	↑ +0.0103
Recall (Cracked)	0.8962	0.8955	↓ -0.0007
F1-score (Cracked)	0.9251	0.9295	↑ +0.0044
Precision (NonCracked)	0.9023	0.9027	↔ roughly same
Recall (NonCracked)	0.9587	0.9686	↑ +0.0099

Table 5: Confusion-matrix comparison for the balanced training runs

Category	10 Epochs	20 Epochs	Change
NonCracked → NonCracked	6847	6918	↑ +71
NonCracked → Cracked	295	224	↓ -71
Cracked → Cracked	6400	6395	↓ -5
Cracked → NonCracked	741	746	↑ +5

4.5 Context for Comparison with Public Implementations

Public notebook implementations of SDNET2018 often use different splits, preprocessing steps, architectures, and class distributions. Their reported accuracy values are therefore not directly comparable with the present results unless the evaluation protocol is reproduced. The current experiment instead demonstrates within one workflow that balancing changes the interpretation of model performance: overall accuracy remained near 93%, while cracked-class recall increased from 59.54% to approximately 89.6%.

V. CONCLUSION

This study evaluated a ResNet18 classifier for binary crack detection using imbalanced and balanced versions of SDNET2018. The imbalanced model achieved 93.02% overall accuracy but detected only 59.54% of cracked images. After class balancing, cracked-class recall increased to 89.62% after 10 epochs, with an overall accuracy of 92.75%. These findings show that overall accuracy alone can provide a misleading assessment when non-cracked images dominate the dataset.

Training the balanced model for 20 epochs increased overall accuracy to 93.21% and cracked-class F1-score to 92.95%. However, cracked-class recall remained nearly unchanged at 89.55%, and the number of missed cracked images increased slightly. The primary observed benefit of longer training was a reduction in false-positive crack predictions rather than improved crack sensitivity.

The results are limited to binary classification on SDNET2018 and should not be interpreted as evidence of field-ready performance. Future work should evaluate the model on independent inspection images, report performance separately for decks, walls, and pavements, and test robustness to lighting changes, blur, stains, occlusion, and narrow cracks. Additional work should also examine multi-class defect severity, pixel-level localization, calibration of prediction confidence, and integration with unmanned aerial or robotic inspection systems.

REFERENCES

- [1] Cardellicchio, A., Ruggieri, S., Nettis, A., Patruno, C., Uva, G., and Renò, V. (2022). Deep learning approaches for image-based detection and classification of structural defects in bridges. In *Image Analysis and Processing: ICIAP 2022 Workshops, Lecture Notes in Computer Science*, 13373, 269–279. https://doi.org/10.1007/978-3-031-13321-3_24
- [2] Chun, P. J., Yamane, T., and Maemura, Y. (2022). A deep learning-based image captioning method to automatically generate comprehensive explanations of bridge damage. *Computer-Aided Civil and Infrastructure Engineering*, 37(11), 1387–1401.
- [3] Maguire, M., Dorafshan, S., and Thomas, R. J. (2018). SDNET2018: A concrete crack image dataset for machine learning applications. Utah State University. <https://doi.org/10.15142/T3TD19>
- [4] Swamy et al. (2024). Bridge inspection using image enhancement and machine learning. In *2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science*.
- [5] Zhang, C., Lei, X., and Xia, Y. (2025). Enhancing bridge inspection data quality using machine learning. *Automation in Construction*, 175. <https://doi.org/10.1016/j.autcon.2025.106182>